

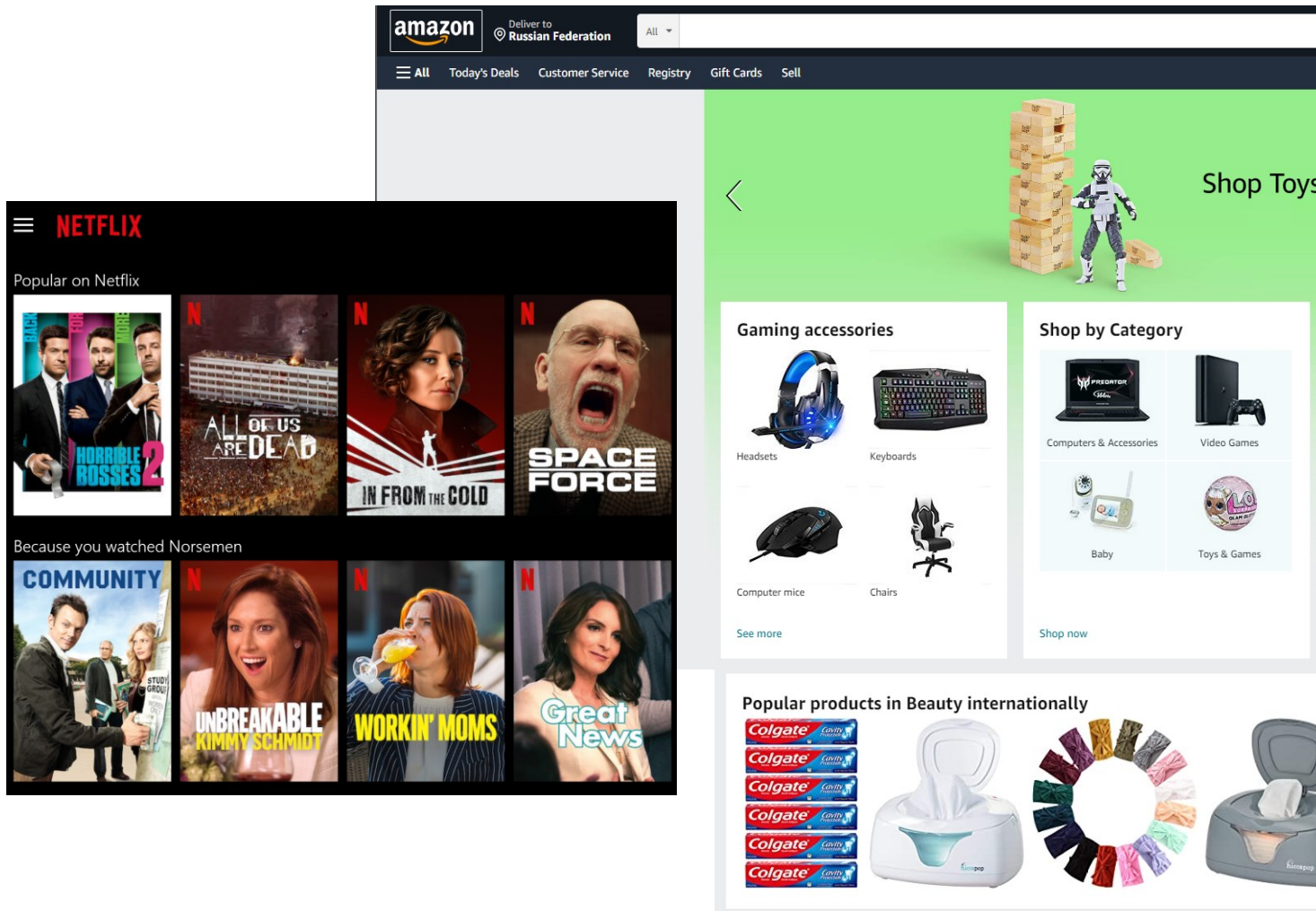
Recommender Systems

Lecture 2

Today's Lecture

- Popularity-based models
- Content-based recommendations
- Baseline Predictors

Usecases for popularity-based predictions



Hacker News new | past | comments | ask | show | jobs | submit

1. **Facebook loses users for the first time** (washingtonpost.com)
1076 points by prostoalex 11 hours ago | hide | 801 comments
2. **In second largest DeFi hack, Blockchain Bridge loses \$320M Ether** (blockworks.co)
431 points by CharlesW 10 hours ago | hide | 359 comments
3. **Squabble.me – Multiplayer Wordle Royale** (squabble.me)
40 points by gundmc 2 hours ago | hide | 18 comments
4. **Optimizing GoAWK with a bytecode compiler and virtual machine** (benhoyt.com)
16 points by benhoyt 1 hour ago | hide | discuss
5. **Starlink adds another 100Gbps connection to the SeattleIX** (seattleix.net)
195 points by walrus01 10 hours ago | hide | 123 comments
6. **Settings are not a design failure** (linear.app)
419 points by tommoor 17 hours ago | hide | 292 comments
7. **Rustenstein 3D: Game programming like it's 1992** (nextroll.com)
297 points by facundo_olano 15 hours ago | hide | 83 comments
8. **Why isn't there a replication crisis in math?** (jaydaigle.net)
257 points by jseliger 15 hours ago | hide | 209 comments
9. **A Pixel's Color** (freedesktop.org)
67 points by pantalaumon 7 hours ago | hide | 4 comments
10. **Facebook says Apple iOS privacy change will result in \$10B revenue hit this year** (cnbc.com)
56 points by caaqil 2 hours ago | hide | 41 comments

Popularity-based scoring

Task:

- assign a popularity score based on accumulated feedback from users
- mostly based on heuristics
 - e.g., content “interestingness” \leftrightarrow consumption frequency
- typical challenges / issues
 - dynamics and trends
 - e.g., seasonal purchases
 - non-homogeneous distribution
 - e.g., experts vs non-experts
 - attacks and fraud

Examples of user feedback

1. Ratings
2. Upvotes/Downvotes
3. Likes (w/o dislikes)

Таблица сравнения

Grow Food	Performance Food
Официальный сайт: growfood.pro	Официальный сайт: p-food.ru
Рейтинг: 8.77	Рейтинг: 9.59
Просмотров: 127 518	Просмотров: 36 058
Отзывов: 14	Отзывов: 8

Y

Hacker News

[new](#) | [past](#) | [comments](#) | [ask](#) | [show](#) | [jobs](#) | [submit](#)

1.

▲ Seawater electrolysis by adjusting the local reaction environment of a catalyst (nature.com)

99 points by CharlesW 2 hours ago | [hide](#) | 46 comments

2.

▲ Show HN: I trained an AI model on 120M+ songs from iTunes (maroofy.com)

91 points by subtech 2 hours ago | [hide](#) | 60 comments

3.

▲ Play Counter Strike 1.6, with full multiplayer, in the browser (play-cs.com)

951 points by philosopher1234 10 hours ago | [hide](#) | 372 comments

4.

▲ I'm Now a Full-Time Professional Open Source Maintainer (filippo.io)

191 points by chmaynard 4 hours ago | [hide](#) | 29 comments

5.

▲ An obituary for the man who saved North Carolina from Nuclear Disaster (ncrabbithole.com)

75 points by joeatwork 2 hours ago | [hide](#) | 3 comments

Rating-based scoring

Popularity-based recommendations

naïve estimation of item j popularity:

$$\text{score}_{\text{POP}}(j) =$$

- U_j - set of users who rated item j
- r_{ij} - rating of item j provided by user i

Popularity-based recommendations

$$\text{score}_{\text{POP}}(j) = \frac{1}{|U_j|} \sum_{i \in U_j} r_{ij}$$

What are potential flaws?

- insufficient amount of data
 - more data means higher reliability /certainty
 - an item with 100 ratings and average score 4.95 vs item with two 5-star ratings
- different dynamics
 - trends
 - seasonality

In the early days

13.



**SALTON HOUSEWARES, INC.
TR2500C ULTIMATE PLUS
BREAKMAKER**

Buy new: **\$135.99**

In Stock

★★★★★ (1)

14.



**KitchenAid KP26M1XLC
Professional 600 Series 6-Quart
Stand Mixer, Licorice**

Buy new: ~~\$499.99~~ **\$329.99**

10 Used & new from **\$325.00**

Get it by **Monday, Feb 9** if you order in
the next **19 hours** and choose one-day
shipping.

Eligible for **FREE** Super Saver Shipping.

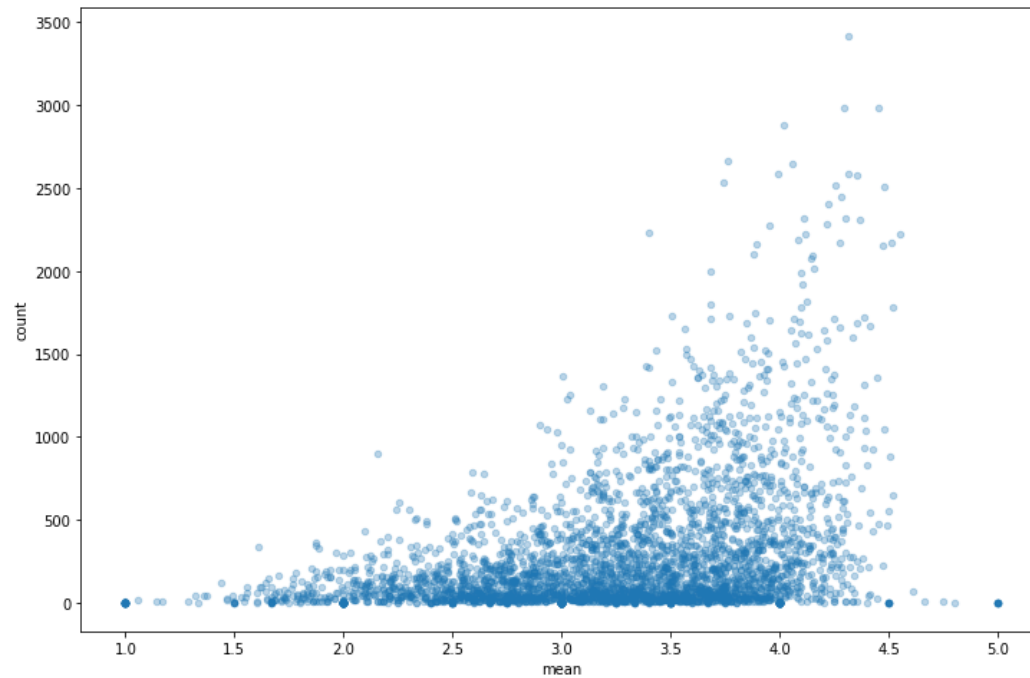
★★★★★ (580)

Image from <https://www.evanmiller.org/how-not-to-sort-by-average-rating.html>

Popular vs niche products

Simple adjustment of popularity

Average rating distribution on Movielens-1M



minimization problem:

Popularity vs trends in data

Bayesian averaging

combination of prior belief and observed sample average:

$$\mu_{BA} = \alpha \cdot \mu_{PB} + (1 - \alpha) \cdot \mu_{SA}, \quad 0 \leq \alpha \leq 1$$

assume we're in the future and already have enough data:

x_k – sample

C – amount of prior samples

n – amount of newly added samples

Bayesian averaging

$$\mu_{BA} = \frac{C\bar{\mu} + \sum_{k=1}^n x_k}{C + n}, \quad \alpha = \frac{C}{C + n}$$

- For item rating data:

$$\mu_j = \frac{C\bar{r}_j + \sum_{i \in U_j} r_{ij}}{C + |U_j|} \quad U_j - \text{set of users, who rated items } j$$

- too high $C \rightarrow$ need more observations
- too low $C \rightarrow$ unreliable estimate

Other possible adjustments

- dampening weights based on recency
- activity of users (bots/trolls or not invested users)
- side information (features, attributes)
 - typically rules-based

Scoring for binary feedback

Possible scoring strategies

- Amount of positive feedback
- $\# \text{ positive feedback} - \# \text{ negative feedback}$
- proportion of positive feedback

In the early days

2. normal

209 up, 50 down 👍👎

A word made up by this corrupt society so they could single out and attack those who are different

Normal is nothing but a word made up by society

conformists worker bees in crowd followers mindless

by Bill Oct 6, 2005 share this add comment

3. normal

118 up, 25 down 👍👎

Urban Dictionary, image credit: <https://www.evanmiller.org/how-not-to-sort-by-average-rating.html>

Estimating popularity on binary feedback

- lower bound of Wilson score confidence interval*
 - estimate for a Bernoulli parameter
 - estimates the bound with 95% probability given current observations

$$\left(\hat{p} + \frac{z_{\alpha/2}^2}{2n} - z_{\alpha/2} \sqrt{\left[\hat{p}(1 - \hat{p}) + \frac{z_{\alpha/2}^2}{4n} \right] / n} \right) / \left(1 + \frac{z_{\alpha/2}^2}{n} \right)$$

- \hat{p} is the observed fraction of positive ratings
- $z_{\alpha/2}$ is the $1 - \alpha/2$ quantile of the standard normal distribution
- n is the total number of ratings

Assumptions: data follows a binomial distribution with

- fixed probability of a success,
- statistically independent trials.

*Example of calculation in python

<https://gist.github.com/adityaookumar/ab81a77258c70f9c3a811f3763a6fb62> ¹⁹

Laplace (add-one) smoothing

- $r_{ij} \in \{0, 1\}$ – binary feedback
upvotes _{j} = $\{r_{ij} > 0, i \in U_j\}$
- $r_{ij} \in \{-1, 0, 1\}$
- $r_{ij} \in \{1, 2, 3, 4, 5\}$

Note on popularity / average ratings

- Horror movies ratings are typically lower, even if a user actually likes it.



- “Ghostbusters” Is A Perfect Example Of How Internet Movie Ratings Are Broken.

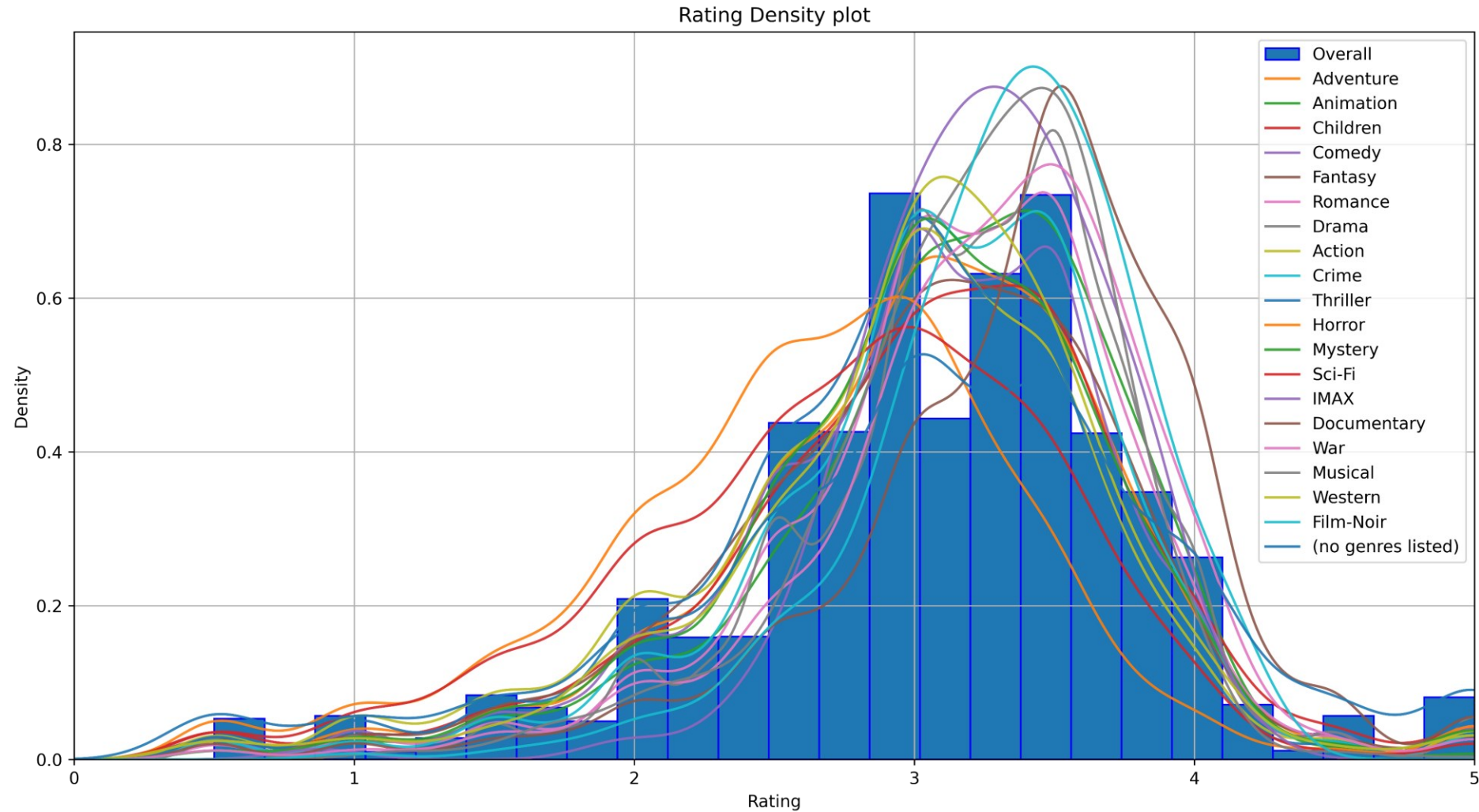


- IMDb **average user rating**: 4.1 out of 10, of 12,921 reviewers
- IMDb **average user rating among men**: 3.6 out of 10, of 7,547 reviewers
- IMDb **average user rating among women**: 7.7 out of 10, of 1,564 reviewers


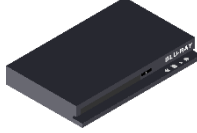

Source: <http://fivethirtyeight.com/features/ghostbusters-is-a-perfect-example-of-how-internet-ratings-are-broken/>

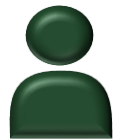
Content-based Recommendations

Ratings distribution over movie genres (ML-25M)

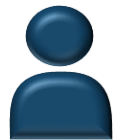


Using content for matching

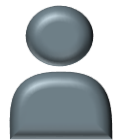
			
Good for gaming	+		+
Good for movies	+	+	+
Good for TV shows	+		
Blue-ray support		+	+



Gamer



Cinema fan



“Family guy”

generating recommendations require:

item features to compute items relevance

user profile information to match items

previous user actions to match items

user attributes for tailoring predictions

criteria specified by user / expert

- knowledge-based
- case-based
- constraint-based

Typical use case: “cold” items

Example of content features

Users

The screenshot shows a Goodreads user profile for 'Steve'. At the top, there's a navigation bar with 'goodreads', 'Home', 'My Books', 'Browse', 'Community', and a search bar. The profile header includes a circular profile picture of a man reading, a 'Follow' button, an 'Add friend' button, and a 'More' dropdown. Below the header, it displays '6,938 pages read' and '19 books read'. A section titled 'STEVE'S FAVORITE BOOKS' shows three book covers: 'The Mindset' by Carol S. Dweck, 'The Power of Habit' by Charles Duhigg, and 'The 7 Habits of Highly Effective People' by Stephen Covey. Another section shows 'Shortest Book' as 'LOVER ED' (228 pages) and 'Longest Book' as 'HARUKI MURAKAMI' (704 pages).

Average book length in 2021 365 pages

- More attributes:
- demographics
 - location
 - occupation
 - ...

Items

The screenshot shows a Goodreads book page for 'Mindset: The New Psychology of Success' by Carol S. Dweck. The book cover is displayed on the left. The title and author are at the top. Below the title, it shows a 4.07 star rating from 114,966 ratings and 8,873 reviews. A green 'Want to Read' button is visible. The description states: 'After decades of research, world-renowned Stanford University psychologist Carol S. Dweck, Ph.D., discovered a simple but groundbreaking idea: the power of mindset. In this brilliant book, she shows how success in school, work, sports, the arts, and almost every area of human endeavor can be dramatically influenced by how we think...more'. There are buttons for 'Amazon' and 'Stores'. A 'FRIEND REVIEWS' section is partially visible. On the right, there's a 'READERS ALSO ENJOYED' section with book covers for 'Atomic Habits', 'GRIT', and 'The Power of Habit'. Below that is a 'GENRES' table.

GENRES	
Psychology	2,453 users
Nonfiction	2,158 users
Self Help	1,469 users
Business	737 users
Self Help > Personal Development	646 users
Education	597 users
Leadership	328 users
Parenting	327 users
Productivity	177 users
Teaching	173 users

- Other features:
- price
 - format/style
 - language
 - ...

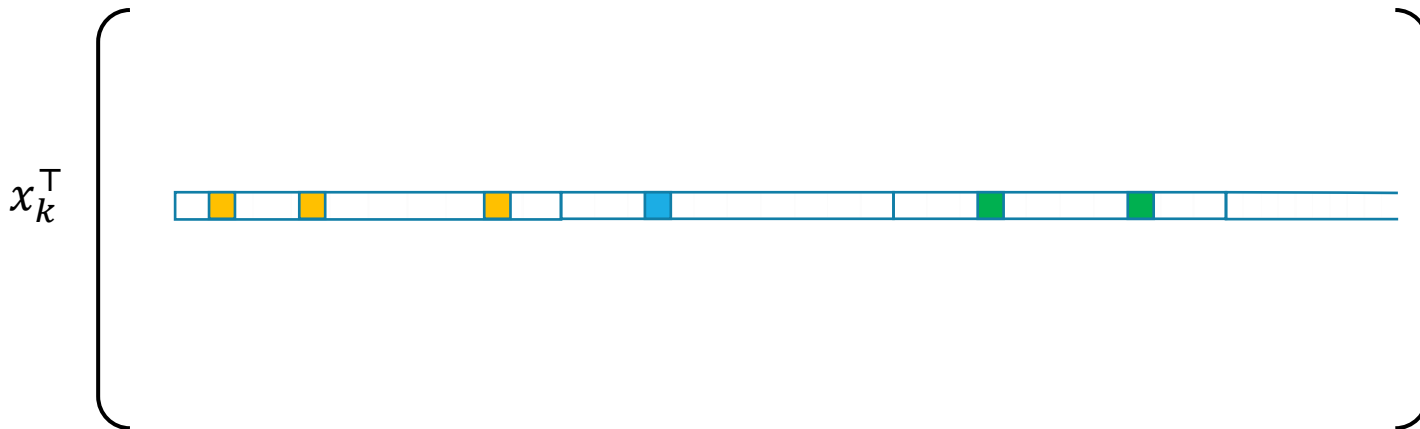
Simple rating prediction

Task: find a utility function f :

$f(\text{item features}) \rightarrow \text{feedback}$

Linear regression model: $X\mathbf{w} = \mathbf{y}$

Linear w.r.t. what?



Linear Regression

$$\mathbf{y} = X\mathbf{w} + \boldsymbol{\epsilon}$$

$$y_k = \mathbf{x}_k^\top \mathbf{w} + \epsilon_k$$

Under assumption of normally distributed noise,
standard linear least squares problem:

$$\|\mathbf{y} - X\mathbf{w}\|_2^2 \rightarrow \min$$

Multicollinearity problem

Multicollinearity problem

Typical ways to combat the ill-posedness:

- add regularization term
 - e.g., l2-penalty $\lambda \|\mathbf{w}\|^2$
- feature selection
 - hard problem
- feature transformation
 - e.g., dimensionality reduction

Ridge regression

How to select features?

Types of features:

- Known knowns
- Known unknowns
- Unknown unknowns

Adding/omitting regressors:

- Missing important variables → bias
- Adding meaningless variables → overfitting
- No standard “silver bullet” recipe exists.
- Choose the simplest possible model, but not simpler.

Simpson Paradox

Correlation vs causation

Is watching the 1984 Ghostbusters movie killing people?



Age Distribution of UK residents Age 10-59yr
watching 1984 Ghostbusters and 2021 Ghostbusters

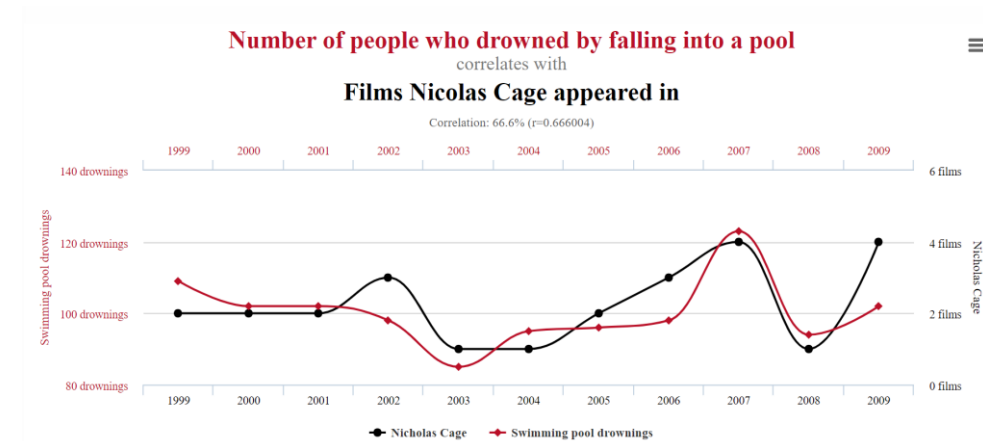
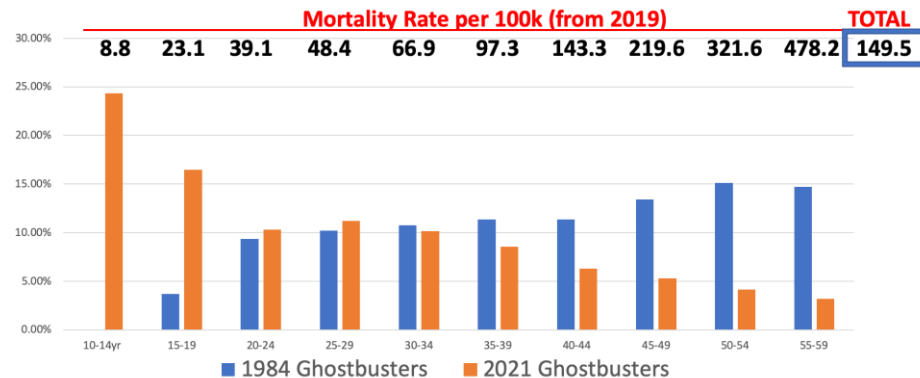


Image from: <https://tylervigen.com/spurious-correlations>

Other methods used for content-based filtering

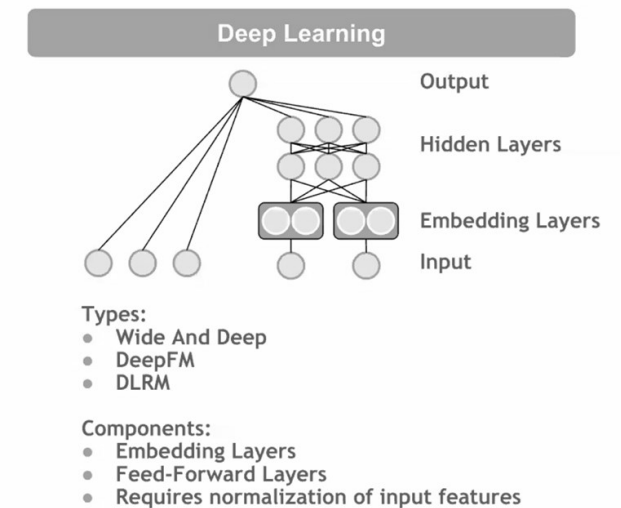
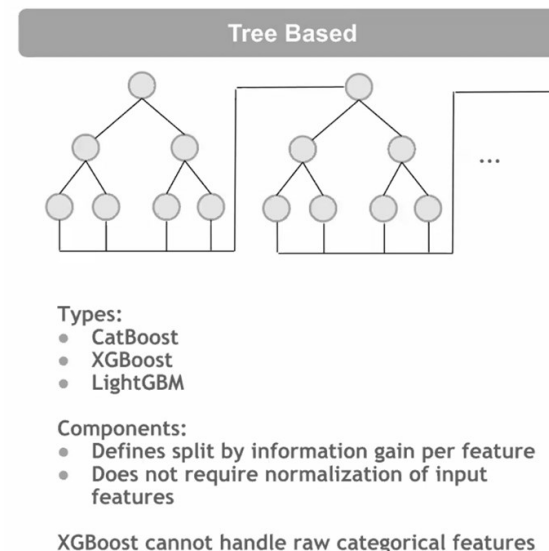
- SVM
- Decision Tree-based models
 - CatBoost
 - LightGBM
 - XGBoost
 - Automated feature selection, e.g. as in LightAutoML
- Neural Networks

Feature engineering usecase

Feature Type	Example	Feature Engineering
Categorical	<ul style="list-style-type: none">User ID / Item IDBrandMain Category	<ul style="list-style-type: none">Target EncodingCount EncodingCategorify + Combining Categories
Unstructured list	<ul style="list-style-type: none">KeywordsSubcategoriesColors	<ul style="list-style-type: none">Target EncodingCount EncodingCategorify
Numeric	<ul style="list-style-type: none">PriceDeliver timeAvg. reviews	<ul style="list-style-type: none">BinningNormalizationGauss Rank
Timestamp	<ul style="list-style-type: none">Timestamp	<ul style="list-style-type: none">Extract month, weekday, weekend, hour
Timeseries	<ul style="list-style-type: none">Events in orderTime since last event	<ul style="list-style-type: none"># of events in past XDifference in time (lag)
Image	<ul style="list-style-type: none">Product image	<ul style="list-style-type: none">Extract latent representation with deep learning
Text	<ul style="list-style-type: none">Description	<ul style="list-style-type: none">Extract latent representation with deep learning
Social graph	<ul style="list-style-type: none">Follower/Following graph	<ul style="list-style-type: none">Link analysis
Geo location	<ul style="list-style-type: none">Addresses	<ul style="list-style-type: none">Distances to point of interest

- Massive experiments in parallel
- Trying, not guessing

Review Recommender Model



How to Build a Winning Deep Learning Recommender System

<https://medium.com/rapids-ai/winning-solution-of-recsys2020-challenge-gpu-accelerated-feature-engineering-and-training-for-cd67c5a87b1f>

https://www.youtube.com/watch?v=bHuww-l_Sqo

Classification

Task: predict whether user will like an item or not.

Example:

$$y_k = \begin{cases} 1, & \text{rating above threshold,} \\ 0, & \text{otherwise.} \end{cases}$$

In linear regression: $y_k \approx \mathbf{x}_k^\top \mathbf{w}$.

How to translate $\mathbf{x}_k^\top \mathbf{w} \in \mathbb{R}$ into binary domain?

Classification – probabilistic formulation

Logistic Regression for classification task

The model:

$$p_{\mathbf{w}}(y_k = 1 \mid \mathbf{x}_k) \sim \frac{1}{1 + e^{-\mathbf{x}_k^{\top} \mathbf{w}}}$$

Target:

$$y_k = \begin{cases} 1, & \text{if } \mathbf{x}_k^{\top} \mathbf{w} > 0, \\ 0, & \text{otherwise.} \end{cases}$$

- Our observations data can be viewed as a set of random trials.
- Outcome of each trial is binary (coin flip).

$$y \sim \text{Bernulli}(\mu)$$

$$E[y] = \mu, \quad 0 \leq \mu \leq 1$$

A bit of probability theory

- We aim to maximize the joint probability distribution (likelihood)

$$p_w(y, X) = \prod_{k=1}^n p_w(y_k, \mathbf{x}_k), \quad p_w(y_k, \mathbf{x}_k) = p_w(y_k \mid \mathbf{x}_k) \cdot p_w(\mathbf{x}_k)$$

$$\operatorname{argmax}_w \prod_{k=1}^n p_w(y_k, \mathbf{x}_k) = \operatorname{argmax}_w \log \prod_{k=1}^n p_w(y_k, \mathbf{x}_k)$$

$$= \operatorname{argmax}_w \left(\log \prod_{k=1}^n p_w(y_k \mid \mathbf{x}_k) + \log \prod_{k=1}^n p_w(\mathbf{x}_k) \right)$$

$$= \operatorname{argmax}_w \sum_{k=1}^n \log p_w(y_k \mid \mathbf{x}_k)$$

Maximum likelihood derivation

$$\mathcal{L}(w) = \sum_{k=1}^n \log p_w(y_k | \mathbf{x}_k)$$

- The probability of each outcome is parametrized via:

$$p_w(y_k | \mathbf{x}_k) = p_w(y = 1 | \mathbf{x}_k)^{y_k} \cdot (1 - p_w(y = 1 | \mathbf{x}_k))^{(1-y_k)}$$

Logistic Regression Objective

More concise/practical form with $y_k \in \{-1, +1\}$ instead of $\{0, 1\}$:

$$\mathcal{L}(\mathbf{w}) = - \sum_{k=1}^n \log \left(1 + e^{-y_k \cdot \mathbf{x}_k^\top \mathbf{w}} \right)$$

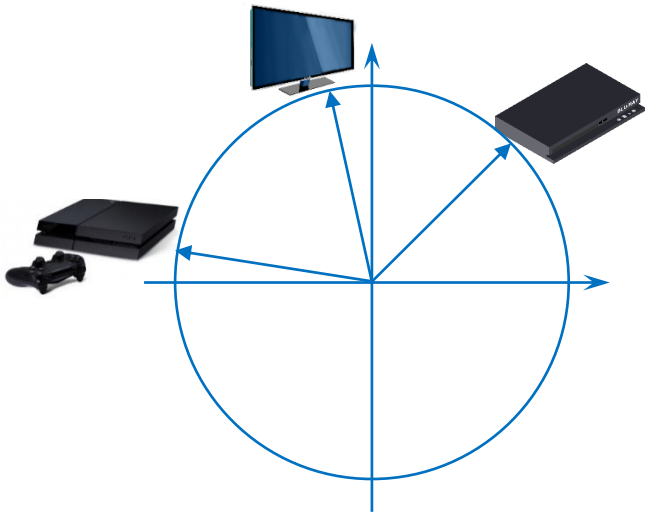
- Turns into standard loss minimization:

$$\mathbf{w}^* = \operatorname{argmin}_{\mathbf{w}} Q(\mathbf{w}), \quad Q(\mathbf{w}) = -\mathcal{L}(\mathbf{w}) + \lambda \|\mathbf{w}\|^2$$

- Can be solved with e.g.:
 - SGD
 - Newton-Raphson (Iteratively Reweighted Least Squares)

Similarity-based models

Each item as a vector in d -dimensional feature space.



Given two item feature vectors \mathbf{x}_i and \mathbf{x}_j , we can estimate their proximity in on a unit sphere:

$$\cos(\alpha) = \frac{\mathbf{x}_i^\top \mathbf{x}_j}{\|\mathbf{x}_i\| \cdot \|\mathbf{x}_j\|}$$

What happens if some features are missing?

How to match with a user?



User profile:

$$\bar{\mathbf{x}}_u = \sum_{j \in I_u} w_{uj} \mathbf{x}_j, \quad I_u - \text{set of items of user } u$$

The simplest case:

$$w_{uj} = \frac{1}{|I_u|} \quad \text{or} \quad w_{uj} = \frac{r_{uj}}{\sum_k r_{uk}},$$

Relevance estimation:

$$r_{ui} = \frac{\bar{\mathbf{x}}_u^\top \mathbf{x}_i}{\|\bar{\mathbf{x}}_u^\top\| \cdot \|\mathbf{x}_i\|}$$

How to evaluate

Intuitive heuristic:

- Based on proximity to user preferences:
 - must be close to what user likes
 - must be distant from what user dislikes
- Construction of evaluation dataset:
 - select k pairs of likes/dislikes for each user
 - must be excluded from training data
 - rank other items (e.g., user favorites) based on their proximity to test pairs