IEOR E4650 Business Analytics

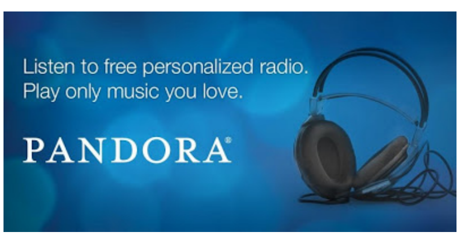
# Session 9: Recommendation Analytics
# Pandora

Spring 2018



Prof. Adam Elmachtoub

## What is Pandora and What Value Does it Capture?




- Internet radio station featuring personalized playlist tailored to user's taste
- Tim Westergren: founder of Pandora, was Chief Strategy Officer and now CEO again
- Very successful in the market: about 80 million active users; over 50 billion thumb up's

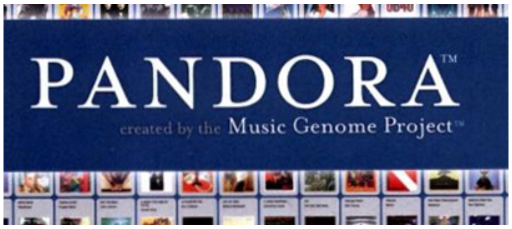
## Outline

- Recommendation Systems
- Quantifying the unquantifiable: Content-based vs. user-based recommendations
- How did Pandora capture \$3.14 billion (today's market cap) in value through analytics?
- Predictive analytic technique: $k$ nearest neighbors ($k$-NN)



## How Does Pandora Do It?

## The Music Genome Project



- Conceived by Will Glaser and Tim Westergren in 1999; capture the essence of music at a fundamental level
- 5 genomes: pop/rock, hip-hop/electronica, jazz, world music, and classical
- Categories of attributes: melody, harmony, rhythm, form, sound (i.e., instrumentation and voice), lyrics
- Specific attributes (rated by analysts on a 0 to 5 scale)
  - Acid rock qualities, accordion playing, acousti-lectric sonority, acousti-synthetic sonority, . . .
- Example: For Led Zeppelin's song "Kashmir," the rating starts 4-0-3-3 (high on acid rock attributes, no accordion, medium sonorities)



## Music Genome Project: The Main Challenge is . . .

. . . creating the data!

How many songs can be rated in nine months? What is the cost?

- 450 musical attributes (250 attributes for a pop song)
- 50 song analysts; 20 minutes for one analyst to rate a pop song on 10 attributes
- each analyst works 8 hours/day, 20 days/month at 15 $/hour

Number of songs rated in 9 months

- 250 attributes requires 25 analysts working 20 minutes
- 50 analysts can rate 6 songs per hour; 48 songs per day; 960 songs/month
- Approximately 10,000 songs rated in 9 months

Cost

- 50 song analysts; 15 $/hour; 8 hour/day; 20 days/month; 9 months
- $1 million for 9 months to rate 10,000 songs

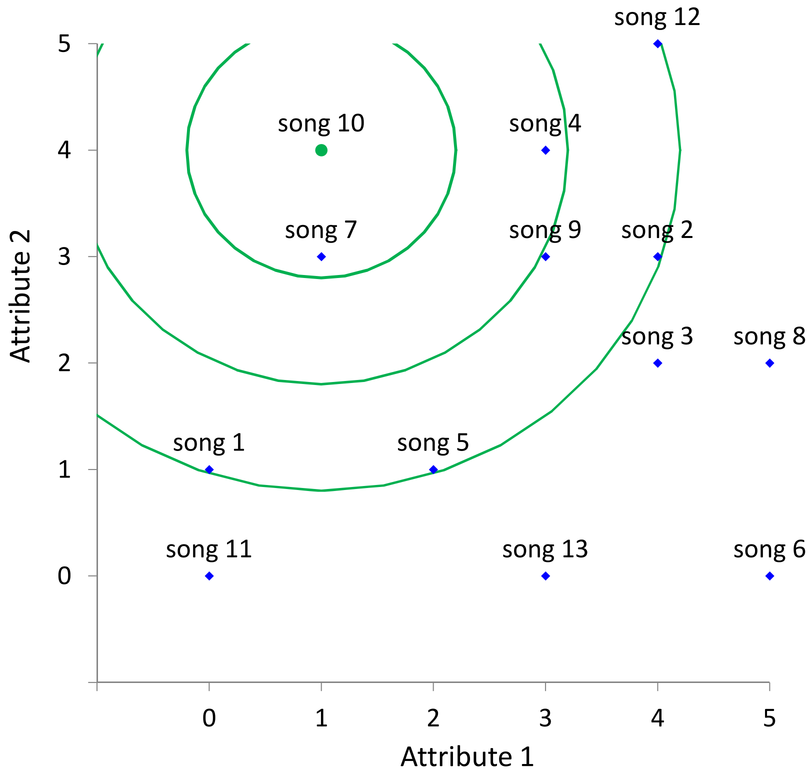
## How Does Pandora Go From Genomes to Recommendations?



## Distance Metrics and the 1-NN Algorithm

- User selects a favorite song
- Find "weighted distance" of favorite song to every other song
- Recommend the song with the minimum weighted distance to the favorite song

# 1-NN Algorithm: Pandora




- A user chooses song 10 to listen to first and rates it "like"
  - Assume the two attributes are equally important
- What song would 1-NN pick to play for the user next?
- Song 7: it's the closest to song 10
- What happened in the first test?



# Pandora and Tim Westergren


Link to the full video: http://www.youtube.com/watch?v=1tZiKHVScgg

## Distance

"Distance" between two songs $I$ and $G$:

$$d(I,G) = \sqrt{\sum_i (X_i(I) - X_i(G))^2}$$

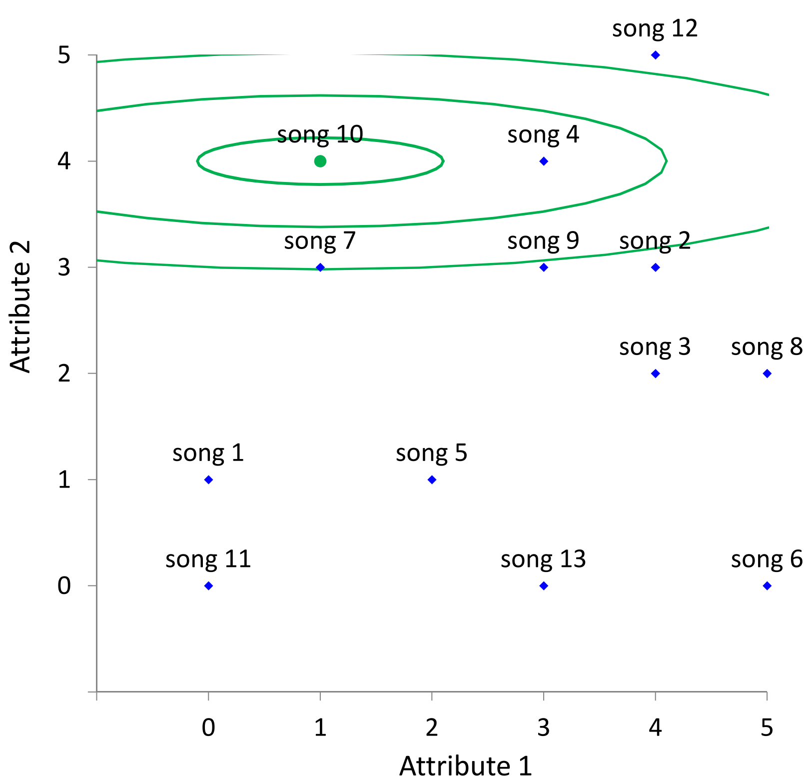
"Weighted distance" between two songs $I$ and $G$:

$$d(I,G) = \frac{1}{\sum_j w_j} \sqrt{\sum_i w_i^2 (X_i(I) - X_i(G))^2}$$

Example: if $w_1 = 1$ and $w_2 = 5$, attribute $2$ is five times as important as attribute $1$



## 1-NN Algorithm: Weighted Distance




- A user chooses song 10 to listen to first and rates it "like"
  - Assume attribute 2 is 5 times as important as attribute 1
- What song would 1-NN pick to play for the user next?
- Song 4 is now the closest to song 10

## Pandora's Patent

Pandora patent link: http://www.google.com/patents/US7003515

"Because not all of the genes are equally important in establishing a good match, the distance is better calculated as a sum that is weighted according to each gene's individual significance."

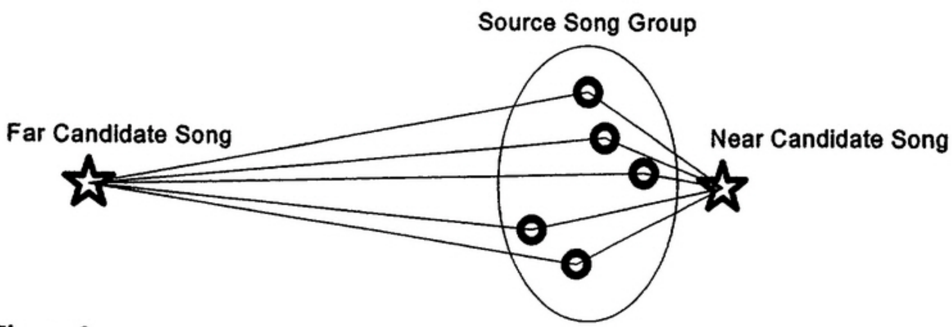



Figure 3



## Applying the Concept of Nearest Neighbors to Prediction

# $k$-NN Algorithm

Predict the response of a new observation based on input variables

- Response: take a loan or not
- Response: like a song or not

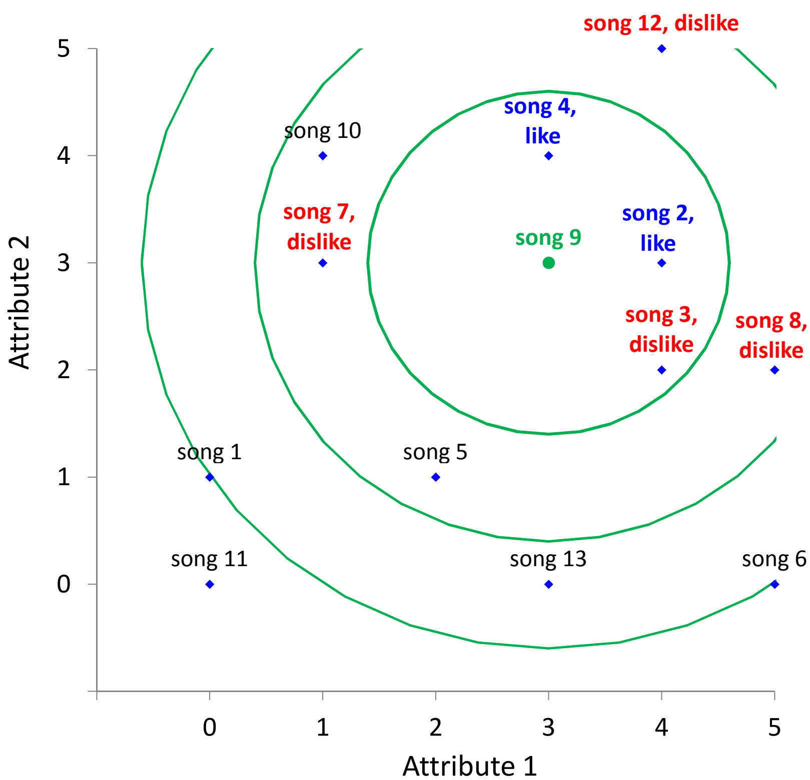
$k$-NN algorithm steps

Step 1. Find the closest $k$ neighbors

Step 2. Predicted response: average response of the $k$ neighbors



# 3-NN Algorithm




- New song: song 9
- Does the 3-NN algorithm predict the user will like or dislike it?
  - Assume attributes are equally important
  - Note: the user has only rated 6 songs
- Three closest neighbors: songs 2 (like), 3 (dislike) and 4 (like)
- 3-NN algorithm: song 9 prediction is 2/3
- 3-NN algorithm: user is more likely to like song 9

## Content-Based Recommendations

Pandora's approach is based on analyzing similarity in content

What are some possible limitations?



## How Else Can We Generate Recommendations?

## Similarity Among Users

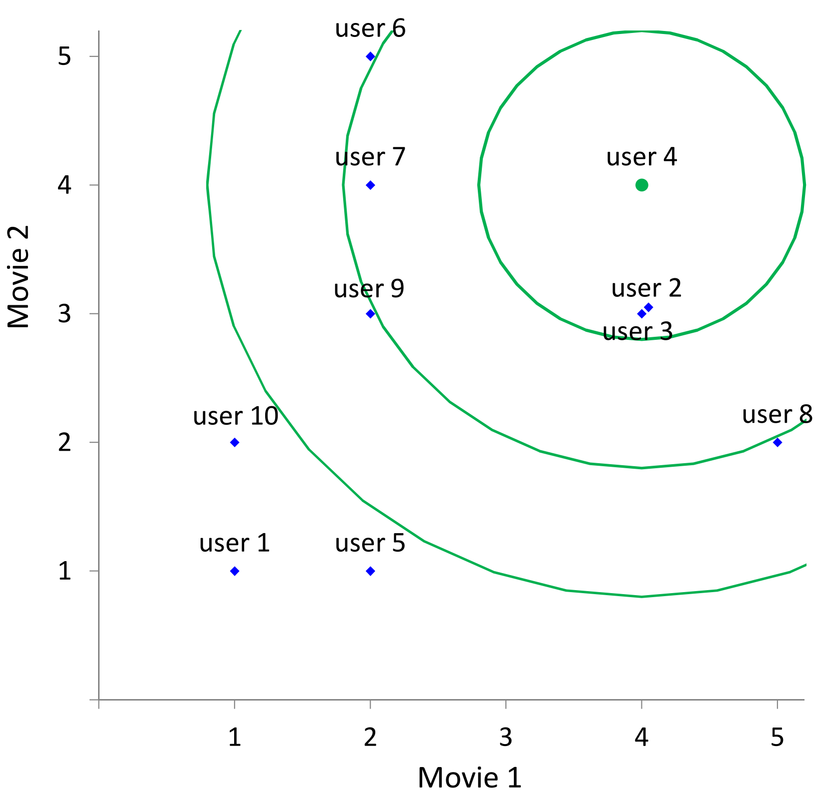
Who shares your taste?

- Data: user ratings of content (e.g., songs, movies, products)
- Compute your “ratings distance” to each of the other users
- Find your nearest neighbor (closest users in “rating space”)
- Form predictions by looking at how your nearest neighbors rated content

Why might this work?



## User-Based $k$-NN Algorithm for Movie Predictions




Goal: make a prediction for user 4 rating movie 3

- Who are the closest users to user 4?
  - In order: user 4, users 2 and 3 (tie), user 7, . . .
- Movie 3 ratings
  - User 2 rates it: 3
  - User 3 rates it: 4
- What movie 3 rating would 2-NN predict for user 4?

  Predicted rating: 3.5

## What Value of $k$ to Use in $k$-NN?

- Small $k$: too "noisy" because the prediction depends on very little information
- Large $k$: too "general" because it predicts the average rating in the population
- Typically some intermediate value is best
- How to find the best?



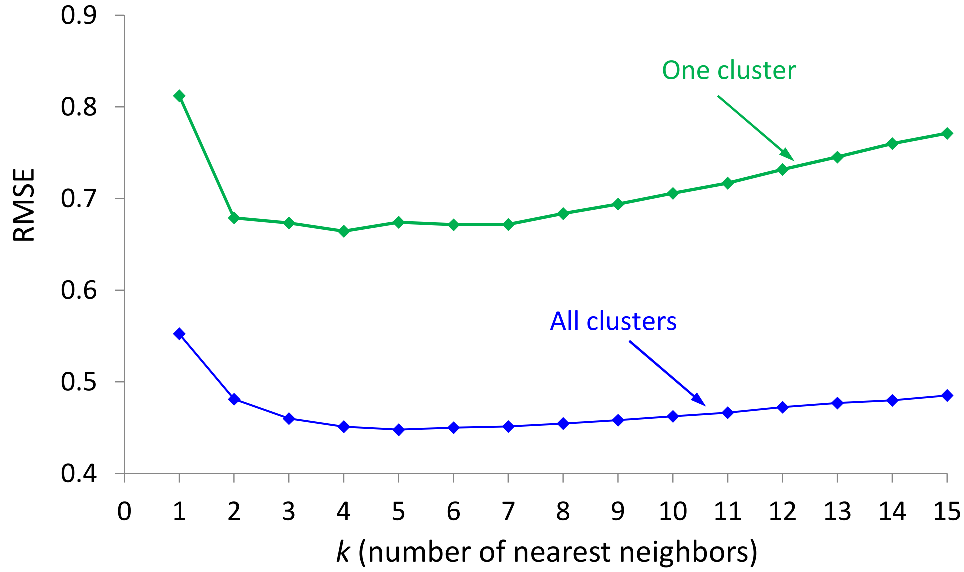
## Find the Best $k$ Using Prediction Error

| Observation | Actual rating (Y) | Prediction | Prediction error |
|---|---|---|---|
| 1 | 5 | 4 | $-1$ |
| 2 | 5 | 4 | $-1$ |
| 3 | 3 | 5 | 2 |
| 4 | 3 | 4 | 1 |

- Actual rating (Y): known movie ratings by users
- Prediction: predicted movie ratings
- Prediction error: prediction minus actual

Aggregate prediction errors for all observations into one number, using Root Mean Square Error (RMSE) as our metric.

# Best $k$ Minimizes Prediction Error (RMSE)




- Repeat the RMSE calculation for different $k$
- The best $k$ is the one with the smallest RMSE
- More data: smaller prediction error (RMSE)



# Netflix Prize

Netflix Prize: $\$1{,}000{,}000$, see http://www.netflixprize.com/

How well did Netflix do before the prize?

> *"The RMSE of Cinematch on the test subset, based on training the Cinematch algorithm using the training data set alone, was* $\mathbf{0.9525}$ *...*
>
> *To qualify for the Grand Prize the RMSE of a Participant's submitted predictions on the test subset must be less than or equal to* $90\%$ *of* $0.9525$*, or* $\mathbf{0.8572}$*."*

Netflix Prize winner: BellKor's Pragmatic Chaos, RMSE $\mathbf{0.8567}$

# Timeline: Part 1

1994 Rolling Stones performance hailed as first major concert broadcast live on the Internet

MIT professor Pattie Maes: website where people can list songs and bands they like

1995 RealAudio introduces streaming Internet audio

Amazon.com goes online; in two months sales are up to $20,000 per week

Broadcast.com: first service to broadcast live internet radio

1996 Artist Formerly Known as Prince bypasses the CD/music store system, sells album directly on the Internet

1997 First personal MP3 player debuts: the Diamond Rio

Netflix launches

1998 U.S. Congress passes DMCA: publishing and performance royalties must be paid for satellite and Internet radio broadcasts (traditional radio broadcasters pay only publishing royalties)

Broadcast.com goes public, sets IPO record when its stock rises 250%

Greg Linden of Amazon applies for a patent on item-to-item collaborative filtering, the "Customers Who Bought This Item Also Bought" function



# Timeline: Part 2

1999 Napster free music file-sharing service launches

Broadcast.com is purchased by Yahoo! for $5.7 billion

Music Genome Project conceived by Will Glaser and Tim Westergren

Pandora founded as Savage Beast Technologies

2000 CD sales peak at nearly $20 billion

Napster has 20 million users

Netflix introduces recommendations with Cinematch

Pandora raises $1.5 million in March from Angel investors just before the tech bubble bursts

Amazon stock falls from $107 to $7

2001 iPod is launched

A court orders Napster to shut down

2002 Westergren files a patent application for his music genome idea

2003 iTunes store opens, sells 70 million songs in its first year at $0.99 each

CD sales slip below $12 billion

2004 Pandora business model switches from technology licensing to Internet radio after broadband penetration grows so that streaming radio can become mass market

2005 Pandora Radio launched

## Timeline: Part 3

2006 Netflix holds an open competition for the best collaborative filtering algorithm to predict user ratings for films

2007 iPhone introduced; iTunes app store goes live

Pandora launches a mobile app on the iPhone the day the app store goes live; number of users spikes from 35,000 to 70,000 per day

2008 Streaming music service Spotify launched

2009 Netflix gives $1 million grand prize to BellKor's Pragmatic Chaos team for improving on Netflix's own algorithm for predicting ratings by more than 10%

2010 Pandora advertising revenue hits $119 million, constituting 87% of total revenue; the rest comes from subscribers who pay a premium for ad-free listening and other perks

2011 Pandora goes public at $16/share, with a $2.6 billion valuation

2012 Pandora reports 125 million registered users in January

2013 Pandora passes 200 million registered users in April, offers about 1 million songs

CD sales continue downward slide; drop to about $3 billion

Apple launches iTunes Radio with tens of millions of songs; Rolling Stone Magazine calls it a Pandora clone

2014 Q4 quarterly revenue is $268 million, of which 78% is from mobile, and 82% from advertising (rest from subscription). 9.82% of all radio listening hours in USA are on Pandora.



## 2015 - OH NO! Spotify is awesome...

## Summary – I

$k$ Nearest Neighbors approach:

- Pros
  - Simple
  - No assumptions on the functional form of the response
  - Can capture unexpected correlations
- Cons
  - Often no easy interpretation of the drivers of the prediction when compared to, e.g., logistic regression
  - More computationally complex
  - Can be difficult to estimate weights



## Summary – II

- Recommendation systems capture value by identifying value-enhancing matches
- Different approaches leverage different data
  - Content-based
  - User-based
- Implications
  - Increase conversion rate
  - Develop demand for "long tail" products
    - At Netflix, two-thirds of the content people watch are recommended by the site
    - More than 90% of its content are watched at least monthly
  - Ease of use (reduce transaction cost)
  - Increase loyalty (lock-in: the more you use it, the better it gets)

# Appendix: Other Recommendation Systems

Amazon: item-to-item collaborative filtering
Amazon's patent link: http://www.google.com/patents/US7113917

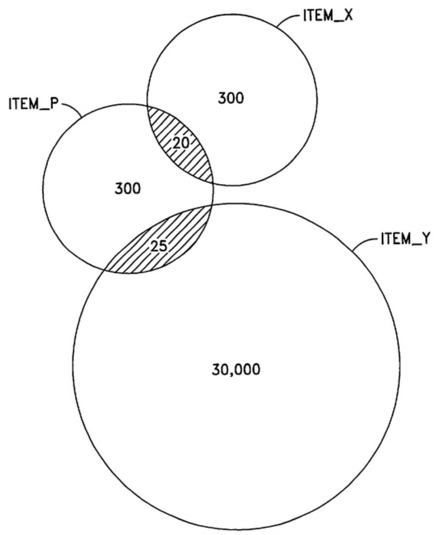



FIG. 4

"Item_P (a popular item) has two 'other items,' item_X and item_Y.

CI(item_P,item_X)=20/sqrt(300x300)=0.0667

CI(item_P,item_Y)=25/sqrt(300x30,000)=0.0083

Thus, even though items P and Y have more customers in common than items P and X, items P and X are treated as being more similar than items P and Y. This result desirably reflects the fact that the percentage of item_X customers that bought item_P (6.7%) is much greater than the percentage of item_Y customers that bought item_P (0.08%)."



# Appendix: Other Recommendation Systems

Google: PageRank algorithm
Google's patent link: http://www.google.com/patents/US6285999

"One aspect of the present invention is directed to taking advantage of the linked structure of a database to assign a rank to each document in the database, where the document rank is a measure of the importance of a document. Rather than determining relevance only from the intrinsic content of a document, or from the anchor text of backlinks to the document, a method consistent with the invention determines importance from the extrinsic relationships between documents. Intuitively, a document should be important (regardless of its content) if it is highly cited by other documents. Not all citations, however, are necessarily of equal significance. A citation from an important document is more important than a citation from a relatively unimportant document. Thus, the importance of a page, and hence the rank assigned to it, should depend not just on the number of citations it has, but on the importance of the citing documents as well. This implies a recursive definition of rank: the rank of a document is a function of the ranks of the documents which cite it. The ranks of documents may be calculated by an iterative procedure on a linked database."

Innovation: deviate from content-based recommendations (initial approach by Yahoo!)