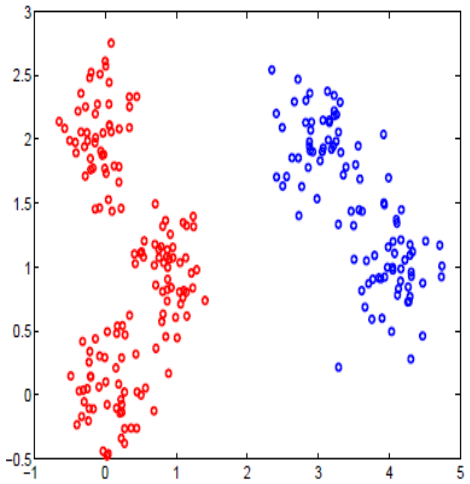
# Data Analysis and Interpretation

# Recap from Week 4

- Cluster: A collection/group of data objects/points
  - similar (or related) to one another within the same group
  - dissimilar (or unrelated) to the objects in other groups

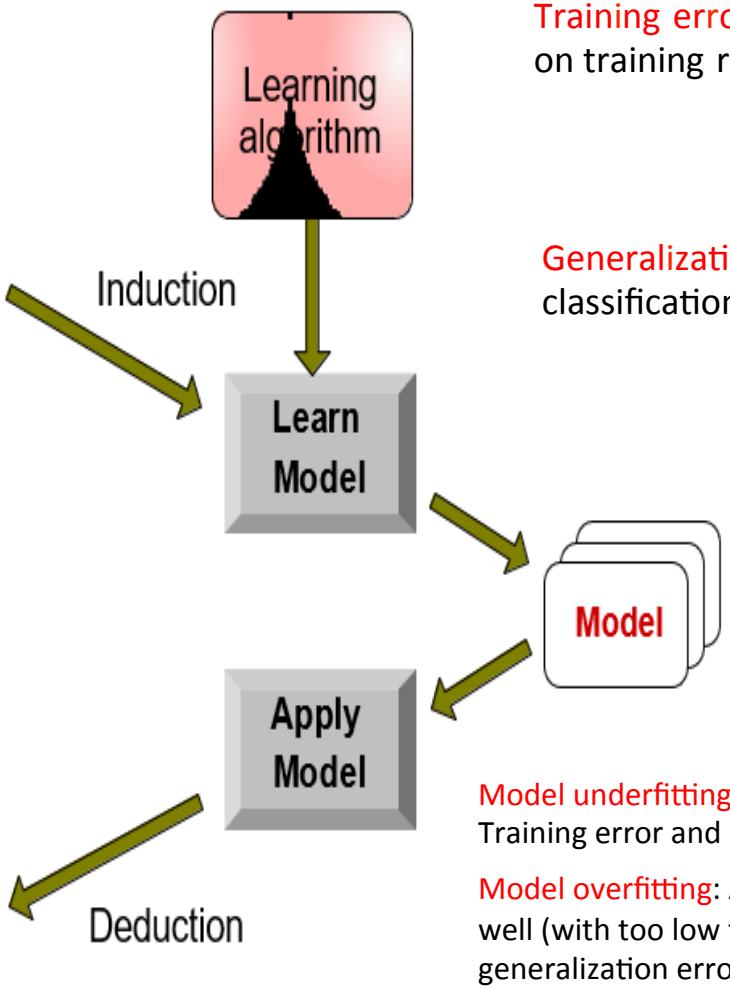


# Recap from Week 5

Classification Pipeline

| Tid | Attrib1 | Attrib2 | Attrib3 | Class |
|---|---|---|---|---|
| 1 | Yes | Large | 125K | No |
| 2 | No | Medium | 100K | No |
| 3 | No | Small | 70K | No |
| 4 | Yes | Medium | 120K | No |
| 5 | No | Large | 95K | Yes |
| 6 | No | Medium | 60K | No |
| 7 | Yes | Large | 220K | No |
| 8 | No | Small | 85K | Yes |
| 9 | No | Medium | 75K | No |
| 10 | No | Small | 90K | Yes |

Training Set




| Tid | Attrib1 | Attrib2 | Attrib3 | Class |
|---|---|---|---|---|
| 11 | No | Small | 55K | ? |
| 12 | Yes | Medium | 80K | ? |
| 13 | Yes | Large | 110K | ? |
| 14 | No | Small | 95K | ? |
| 15 | No | Large | 67K | ? |

Test Set

Training errors: number of classification errors on training records.

Generalization (test) errors: number of classification errors on test records.

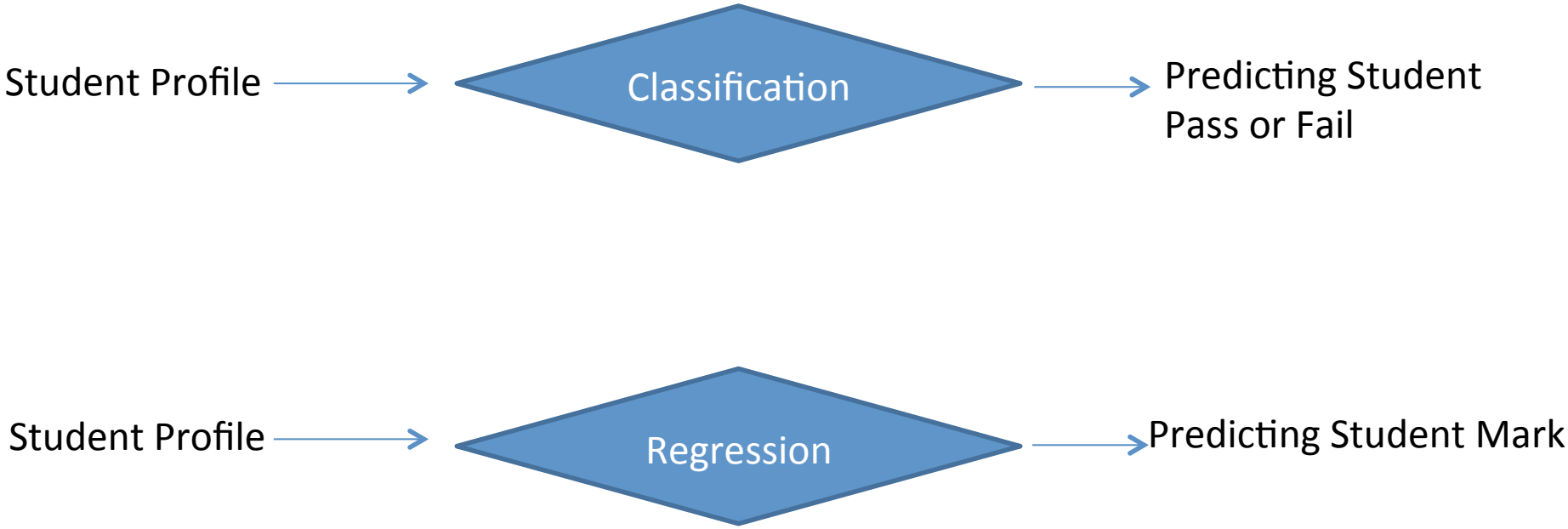
Model underfitting: model is too simple such that both Training error and Generalization error are high.

Model overfitting: A model that fits the training data too well (with too low training errors) may have a higher generalization error than a model with higher training error.

A good model should have low errors of both types.

# Recap from Week 5

- **Classification vs. Regression:**



# 3803ICT course structure

**W1.** Introduction to Data Analytics

Data Preparation and Preprocessing

- **W2.** Data Preparation and Preprocessing

Data Analysis and Interpretation

- **W3.** Exploratory Data Analytics
- **W4.** Statistical Data Analytics
- **W5.** Predictive Data Analytics

Visualization

- **W6.** Data Visualization

Analysis of special types of data

- **W7.** Time Series
- **W8.** Textual Data
- **W9.** Graph Data

Analysis with big data infrastructure

- **W10.** Distributed Data Analysis
- **W11.** Cloud-based Data Analysis

**W12.** Revision

# Predictive Analysis

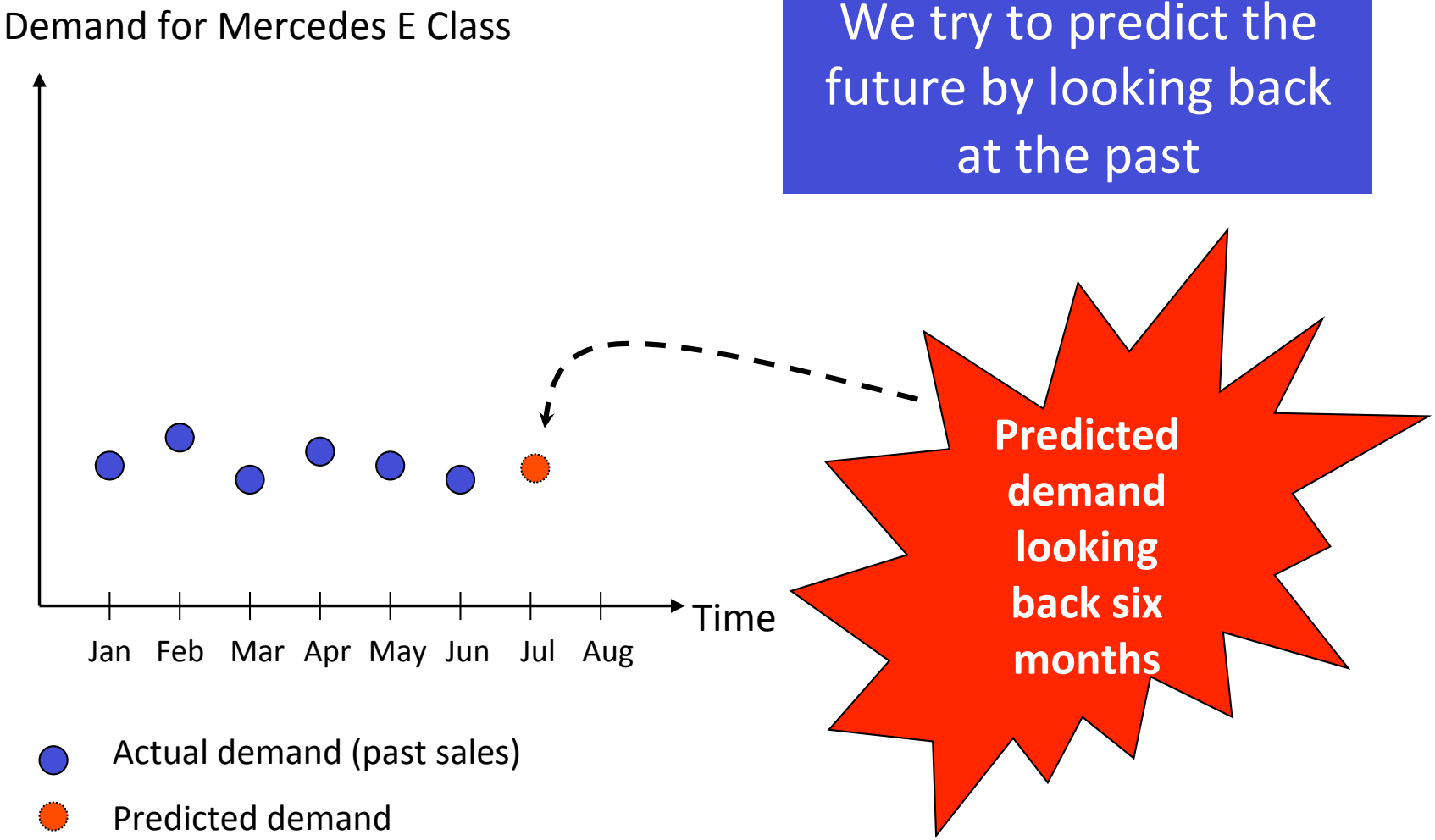
I. **Forecasting:** Temporal Prediction
II. **Recommender Systems:** Personalized Prediction

# I. Forecasting

- What is forecasting?
  - Forecasting is a tool used for predicting future based on past information.
- Applications:
  - Strategic planning (long range planning)
  - Finance and accounting (budgets and cost controls)
  - Marketing (future sales, new products)
  - Production and operations

# Forecasting: Example

We try to predict the future by looking back at the past



# Forecasting: Another Example

| Month | E-class Sales | M-class Sales |
|---|---|---|
| *Jan* | *23,345* | *?* |
| *Feb* | *22,034* | *?* |
| *Mar* | *21,453* | *?* |
| *Apr* | *24,897* | *?* |
| *May* | *23,561* | *?* |
| *Jun* | *22,684* | *?* |
| *Jul* | *23,764* | *?* |

Question: Can we predict the new model M-class sales based on the data in the the table?

# Forecasting: Another Example

| Month | E-class Sales | M-class Sales |
|---|---|---|
| *Jan* | *23,345* | *?* |
| *Feb* | *22,034* | *?* |
| *Mar* | *21,453* | *?* |
| *Apr* | *24,897* | *?* |
| *May* | *23,561* | *?* |
| *Jun* | *22,684* | *?* |
| *Jul* | *23,764* | *?* |

Question: Can we predict the new model M-class sales based on the data in the the table?

Answer: Maybe... We need to consider how much the two markets have in common

# Forecasting Fundamentals

1. Basic Principles
2. Forecasting Process
3. Models
4. Accuracy

# 1. Basic Principles

- **Assumptions:**
  - Historical data contains **some patterns** → support predicting the future
- **Best practices:**
  - Forecasts are rarely perfect
  - Forecasts are more accurate for shorter than longer time periods
  - Forecasts are more accurate for grouped data than for individual items
  - Every forecast should include an error estimate

# 2. Forecasting Process

**Step 1.** Decide what needs to be forecast:

- What is the purpose of the forecast?
- How important is the past in estimating the future?

→ The answers will help determine techniques

**Step 2.** Evaluate and analyze appropriate data

**Step 3.** Select and test the forecasting model

**Step 4.** Apply the forecast on real data

**Step 5.** Monitor forecast accuracy over time

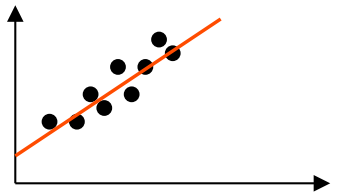
# 3. Forecasting Models

- **Time-series model:** models that predict future demand based on past history trends
- **Causal model:** models that use statistical techniques to establish relationships between various items and demand
- **Simulation:** models that can incorporate some randomness and non-linear effects

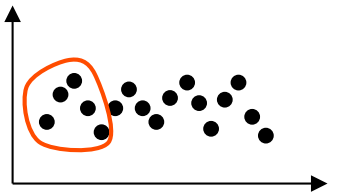
# Forecasting: Time-Series Model

- Forecast based only on past values, no additional information
- Approach: look for data patterns, i.e. data = historical pattern + random variation:

  - **Trends:** persistent, overall upward and downward pattern
    - typically over several years
    - e.g. population, technology, age, culture

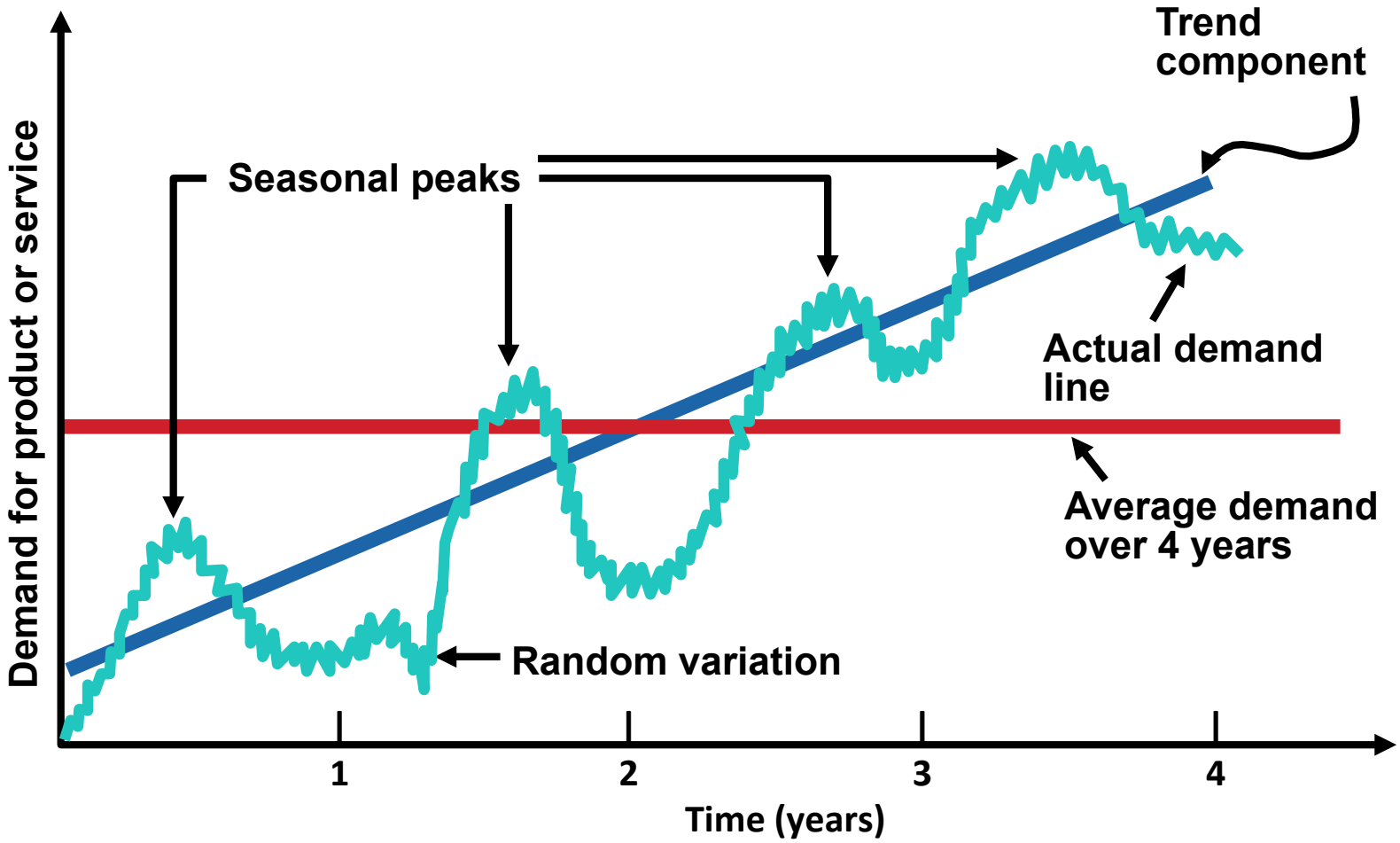


  - **Seasonality:** regular pattern of up and down fluctuations
    - typically over 1 year
    - e.g. weather, customs



  - **Random variation:** erratic, unsystematic, "residual" fluctuations
    - unforeseen events (stocks)
    - Short duration and norepeating

# Historical patterns: Example



# Forecasting: Time-Series Model

- **Techniques:**
  - Moving Average
  - Weighted Moving Average
  - Exponential Smoothing
  - Exponential Smoothing with Trend
  - Seasonality

# Time-Series: Moving Average Method

- Moving Average (MA) is a series of arithmetic means
- **Calculation**: the average value over a set time period (e.g. last four weeks)

$$F_{t+1} = \sum_{i=t-n+1}^{t} A_i / n$$

Where $F_t$ is the forecast value at time t

$A_t$ is the actual value at time t

- **Limitations**:
  - Do not forecast trends well

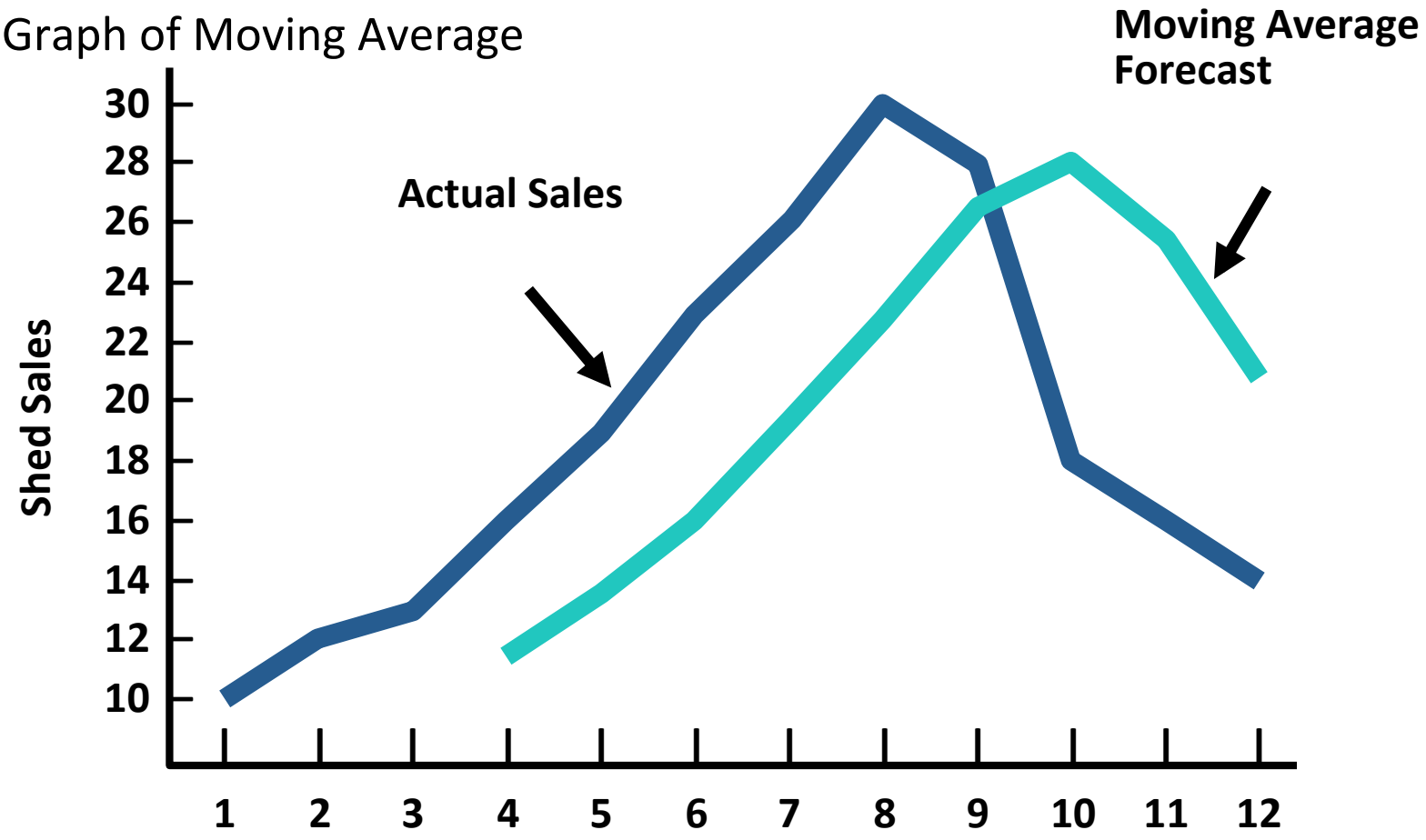
# Time-Series: Moving Average Method

- Example

| Month | Actual Shed Sales | 3-Month Moving Average |
|---|---|---|
| January | 10 | |
| February | 12 | |
| March | 13 | |
| April | 16 | (10 + 12 + 13)/3 = 11 $^{2}/_{3}$ |
| May | 19 | (12 + 13 + 16)/3 = 13 $^{2}/_{3}$ |
| June | 23 | (13 + 16 + 19)/3 = 16 |
| July | 26 | (16 + 19 + 23)/3 = 19 $^{1}/_{3}$ |

# Time-Series: Moving Average Method

- Graph of Moving Average




MV is lagged behind actual data

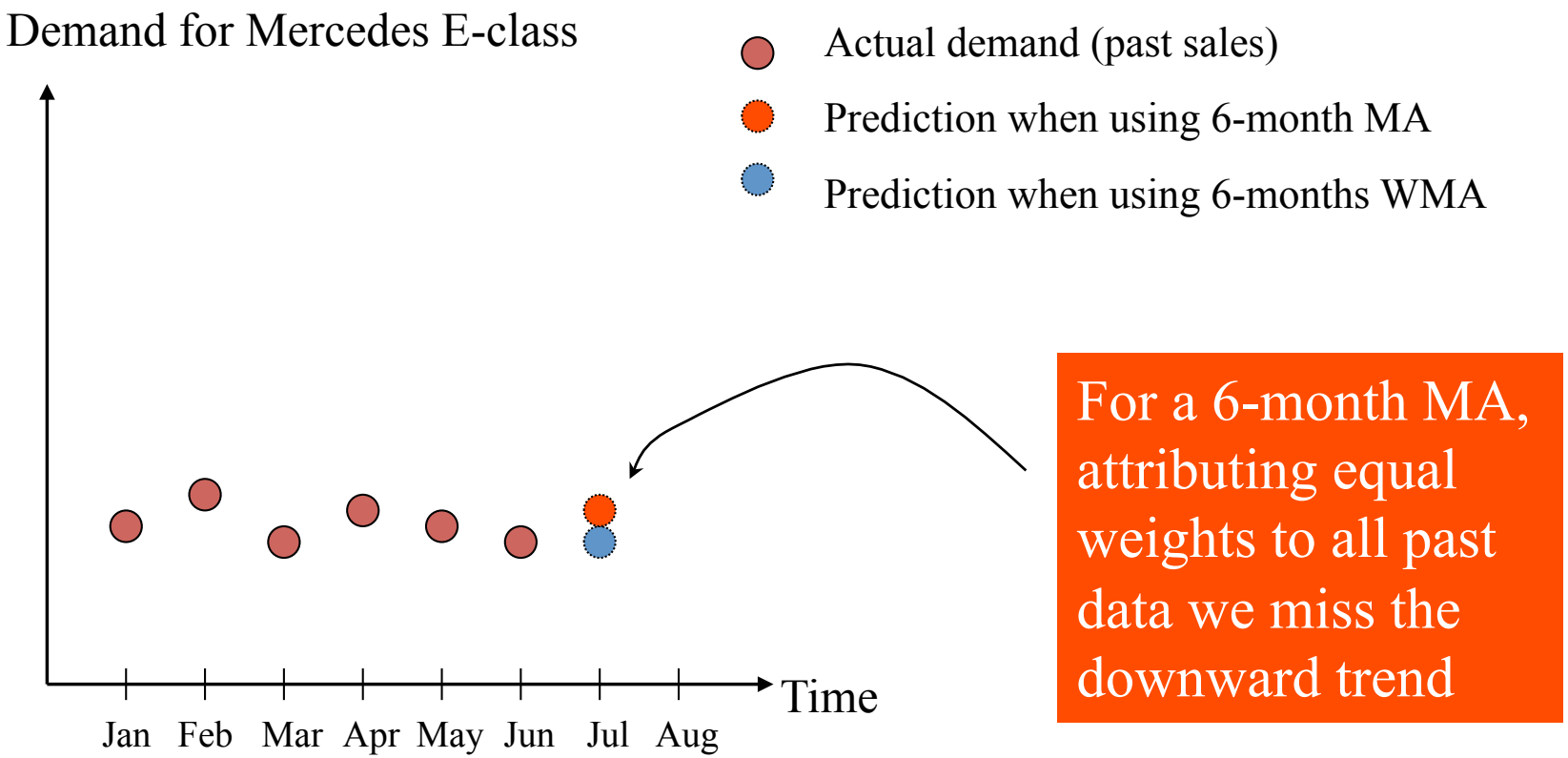
# Weighted Moving Average

- **Limitation of Moving Average:**
  - Weighs all periods equally → less responsive to trends
- **Weighted Moving Average:** used when some trend might be present
  - e.g. older data usually less important
- **Calculation:**

  $F_{t+1} = \sum w_t A_t$

  - All weights must add to 1, e.g. $w_t = 0.5$, $w_{t-1} = 0.3$, $w_{t-2} = 0.2$
    - indicates more weight on recent data
- **Why WMA?**
  - Give more importance to what happened recently, without losing the impact of the past
  - Ability to vary the weights

# Weighted Moving Average: Example




For a 6-month MA, attributing equal weights to all past data we miss the downward trend

# Time-Series: Exponential Smoothing

- Use a weighted combination of last forecast and last actual value

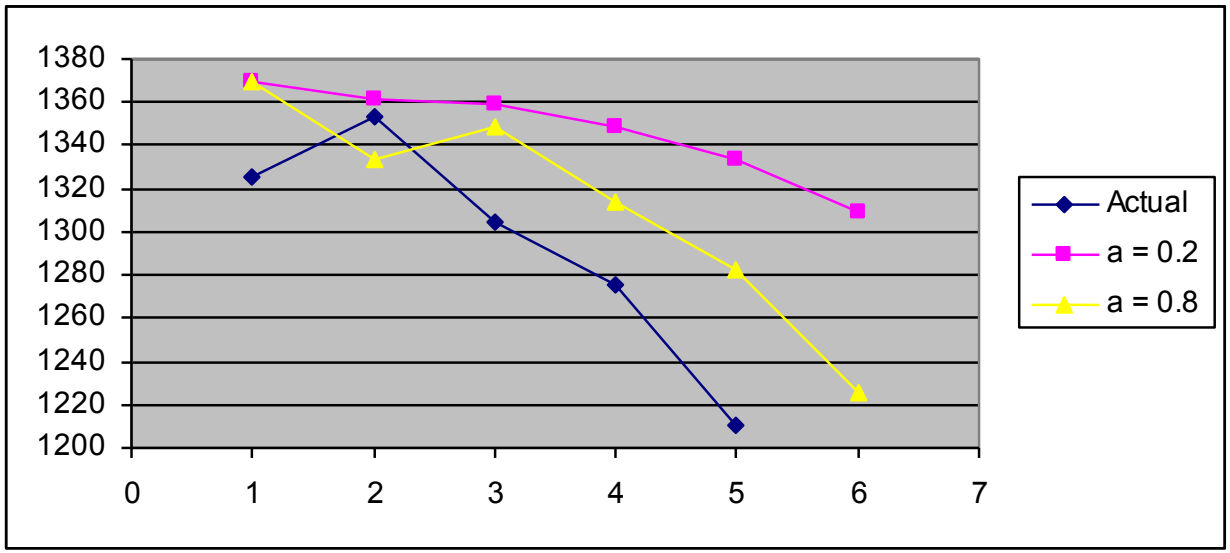
$$F_{t+1} = \alpha A_t + (1-\alpha) F_t$$

$$= \alpha A_t + \alpha(1-\alpha) A_{t-1} + \ldots + \alpha(1-\alpha)^t A_0 \qquad \text{where } \alpha \in [0,1]$$

  - A form of weighted moving average:
    - Most recent data weighted most
    - Weights decline exponentially
- Why exponential smoothing?
  - More responsive to trend
  - Require minimum amount of data needed

# Exponential Smoothing: $\alpha$ trade-off

- Trade-off between trend and random variation: $F_{t+1} = F_t + \alpha(A_t - F_t)$
  - Higher $\alpha$ values (e.g. 0.7 or 0.8) may place too much weight on last period's random variation
    - Good at capturing long-term trend
    - Sensitive to short-term fluctuations



# Time-series: Seasonality

- A university must develop forecasts for the next year's quarterly enrollments. It has collected quarterly enrollments for the past two years. What is the forecast for each quarter of next year?

| Quarter | Year 1 | Year 2 | Year3 |
|---|---|---|---|
| Fall | 24000 | 26000 | ? |
| Winter | 23000 | 22000 | ? |
| Spring | 19000 | 19000 | ? |
| Summer | 14000 | 17000 | ? |

# Time-series: Seasonality

- Step 1. Calculate the average demand for each year
- Step 2. Calculate seasonal indexes
- Step 3. Average the indexes
- Step 4. Forecast demand for the next year
- Step 5. Multiple next year's average seasonal demand by each average seasonal index

| Quarter | Year 1 | Seasonal Index | Year 2 | Seasonal Index | Avg. Index | Year3 |
|---|---|---|---|---|---|---|
| Fall | 24000 | **1.2** | 26000 | **1.24** | **1.22** | **26840** |
| Winter | 23000 | **1.15** | 22000 | **1.05** | … | **…** |
| Spring | 19000 | **0.95** | 19000 | … | | |
| Summer | 14000 | **0.7** | 17000 | … | | |
| Average | **20000** | | **21000** | | | **22000** |

# Time-series: Seasonality

- The multiplicative seasonal model can adjust trend data for seasonal variations in demand (jet skis, snow mobiles)

- **Steps:**
  1. Calculate the average demand per season
     - E.g.: average quarterly demand
  2. Calculate a seasonal index for each season of each year:
     - Divide the actual demand of each season by the average demand per season for that year
  3. Average the indexes by season
     - E.g.: take the average of all Spring indexes, then of all Summer indexes, ...
  4. Forecast demand for the next year & divide by the number of seasons
     - Use regular forecasting method & divide by four for average quarterly demand
  5. Multiply next year's average seasonal demand by each average seasonal index
     - Result is a forecast of demand for each season of next year

# Forecasting: Causal Model

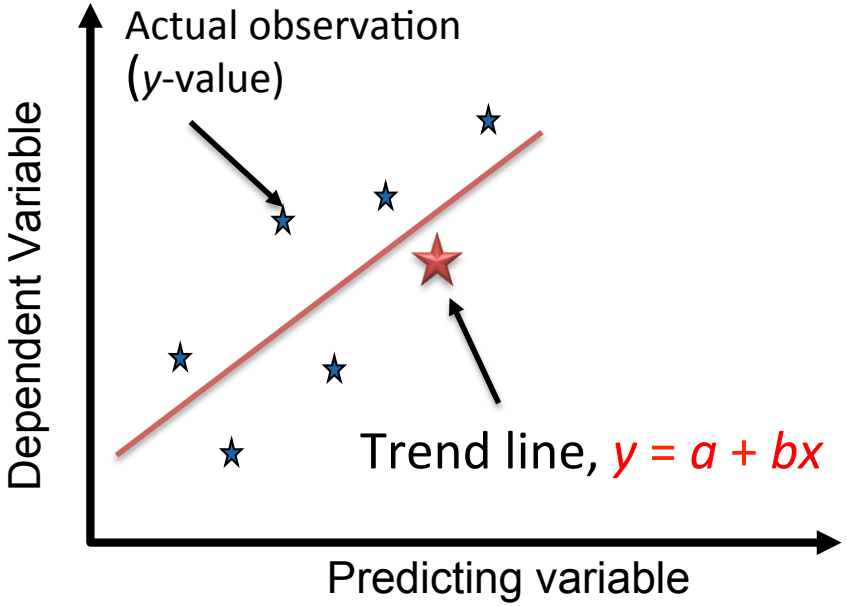
- Causal models establish a cause-and-effect relationship between independent and dependent variables

*A maker of golf shirts has been tracking the relationship between sales and advertising dollars. Use linear regression to find out what sales might be if the company invested $53,000 in advertising next year.*

| | Sales $ (Y) | Adv.$ (X) |
|---|---|---|
| 1 | 130 | 32 |
| 2 | 151 | 52 |
| 3 | 150 | 50 |
| 4 | 158 | 55 |
| 5 | ? | 53 |

# Causal Model: Problem

- Fitting a trend line to historical data points to project into the future (medium to long-range)
- Linear trends can be found using the least squares technique




Infer the regression variables:

$$b=\sum xy-nx\,y\,/\sum x\uparrow 2-nx\uparrow 2$$

$$a=y-bx$$

# Linear Regression: Example

- A maker of golf shirts has been tracking the relationship between sales and advertising dollars. Use linear regression to find out what sales might be if the company invested $53,000 in advertising next year.

| | Sales $ (Y) | Adv.$ (X) | XY | $X^2$ | $Y^2$ |
|---|---|---|---|---|---|
| 1 | 130 | 32 | 4160 | 2304 | 16,900 |
| 2 | 151 | 52 | 7852 | 2704 | 22,801 |
| 3 | 150 | 50 | 7500 | 2500 | 22,500 |
| 4 | 158 | 55 | 8690 | 3025 | 24964 |
| 5 | **153.85** | 53 | | | |
| Sum | 589 | 189 | 28202 | 9253 | 87165 |
| Avg | **147.25** | **47.25** | | | |

$$b = \frac{\sum XY - n\bar{X}\bar{Y}}{\sum X^2 - n\bar{X}^2}$$

$$b = \frac{28202 - 4(47.25)(147.25)}{9253 - 4(47.25)^2} = 1.15$$

$$a = \bar{Y} - b\bar{X} = 147.25 - 1.15(47.25)$$

$$a = 92.9$$

$$Y = a + bX = 92.9 + 1.15X$$

$$Y = 92.9 + 1.15(53) = 153.85$$

# Forecasting: Evaluation

- Mean Absolute Deviation (MAD)
  - Measures the total error in a forecast without regard to sign
  - Higher MAD implies worse performance

$$\mathbf{MAD} = \frac{\sum \left| \mathbf{actual - forecast} \right|}{\mathbf{n}}$$

- Mean Square Error (MSE)
  - Penalizes larger errors

$$\mathbf{MSE} = \frac{\sum \left( \mathbf{actual - forecast} \right)^2}{\mathbf{n}}$$

# Evaluation: Guideline on how to select the right forecasting model

1. The amount & type of available data
   - Some methods require more data than others
2. Degree of accuracy required
   - Increasing accuracy means more data
3. Length of forecast horizon
   - Different models for 3 month vs. 10 years
4. Presence of data patterns

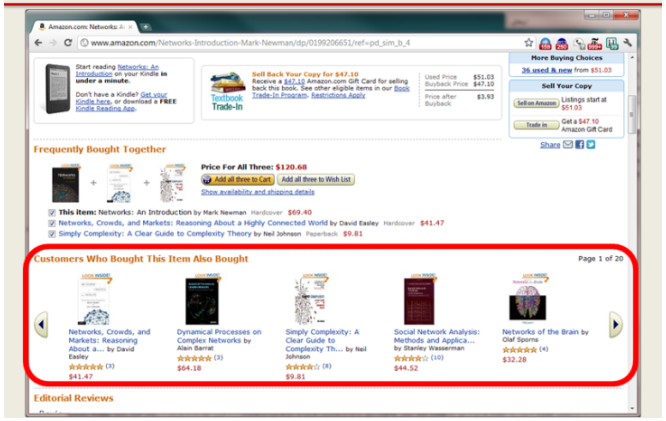
# II. Recommendation Systems

- Which mobile phone should I buy?
- Where should I visit for my business trip?
- Whom should I follow on Twitter?
- Where should I invest my money?

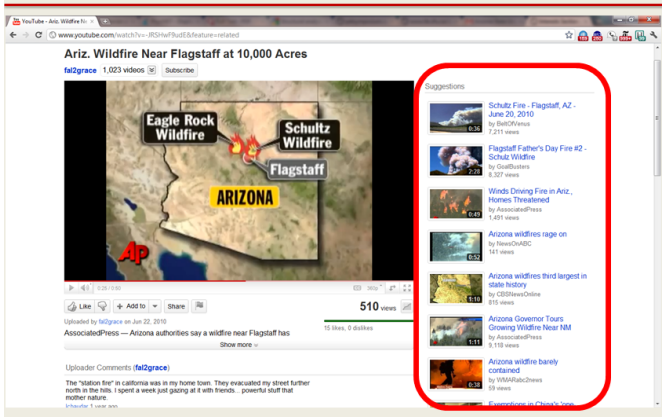
- Which tour is the best for **our class**?

# Recommender Systems - Applications

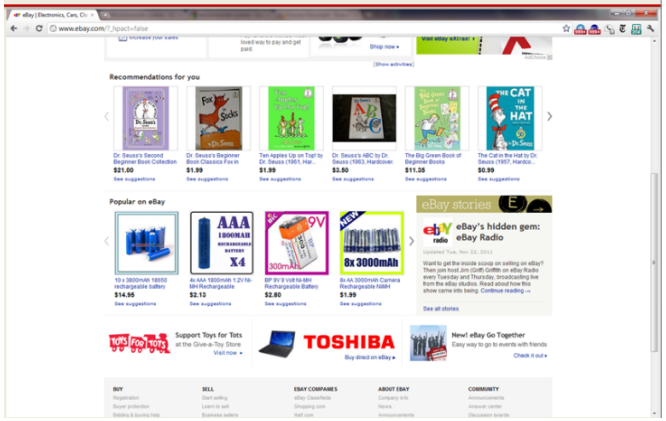
## Book recommendation in Amazon



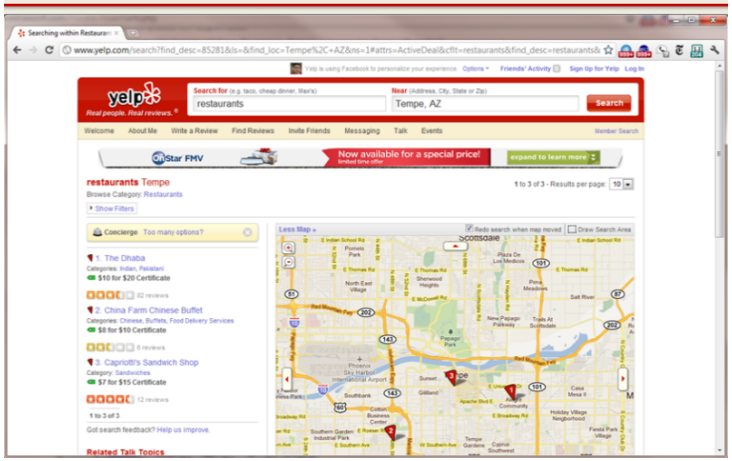

## Video clip recommendation in YouTube




## Product Recommendation in ebay



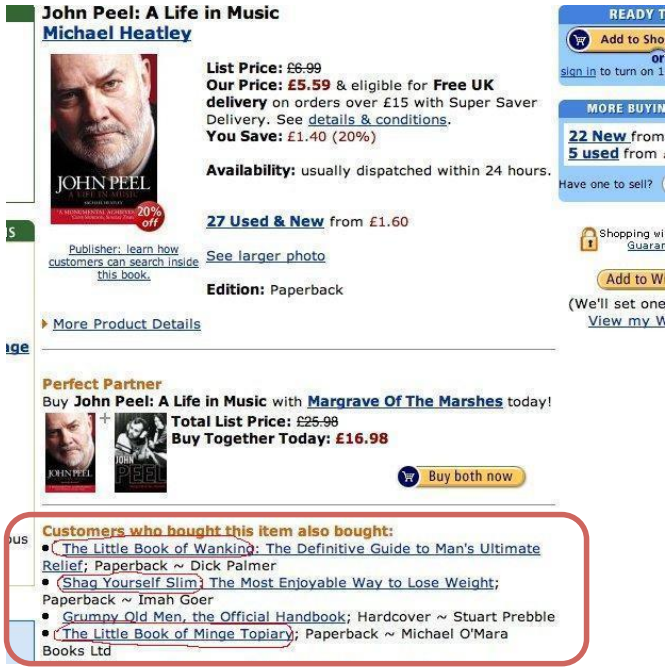

## Restaurant Recommendation in Yelp




recommendation = personalized prediction

# The value of recommendations

- Netflix: 2/3 of the **movies** watched are recommended
- Google News: recommendations generate 38% more click through
- Amazon: 35% **sales** from recommendations
- Choicestream: 28% of the people would buy more **music** if they found what they liked.




# Recommender Systems - Problem

- Estimate a utility function that automatically predicts how a user will like an item
  - Formally, a recommender system takes a set of users $U$ and a set of items $I$ and learns a function $f$ such that: $f: U \times I \rightarrow R$
  - For each user $u \in U$, we want to choose an item $i \in I$ that maximize $f$.
- Based on:
  - **Past behavior**
  - **Relations to other users**
  - **Item similarity**
  - **Context**
  - …

# Recommender Systems - Challenges

- Cold-Start Problem
  - Recommender systems use **historical data** or information provided by the user to recommend items, products, etc.
  - When user join sites, they still haven't bought any product, or they have no history.
  - It is hard to infer what they are going to like when they start on a site.
- Data Sparsity
  - When historical or prior information is **insufficient**.
  - Unlike the cold start problem, this is in the system as a whole and is not specific to an individual.
- Attacks:
  - **Push Attack**: pushing ratings up by making fake users
  - **Nuke attack**: DDoS attacks, stop the whole recommendation systems
- Explanation
  - Recommender systems often recommend items with **no explanation** on why these items are recommended

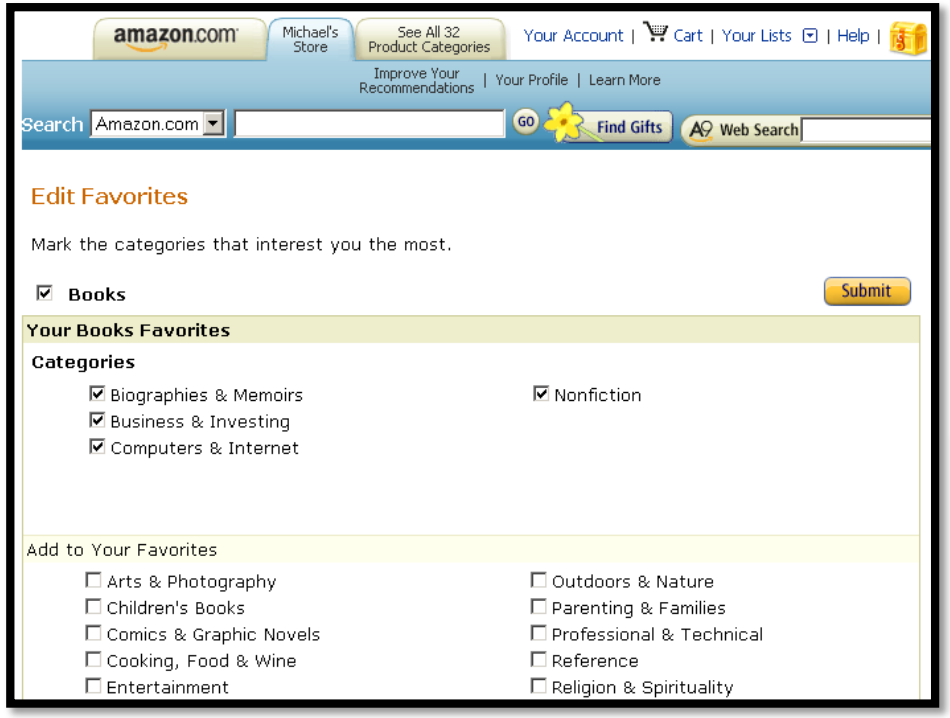
# Types of Recommender Systems

1. Content-based recommendation:
   - Recommend **based on similarity** between user features and item features
2. Rating-based recommendation (Collaborative Filtering)
   - Recommend **based on rating** matrix
3. Clustering-based recommendation
   - Recommend **based on clusters of rating** matrix

# 1. Content-based recommendation

- Assumption: a **user's interest** should match the description of the items that the user should be recommended by the system.
  - The more similar the item's description to that of the user's interest, the more likely that the user finds the item's recommendation interesting.
- Goal: **find the similarity** between the user and all of the existing items is the core of this type of recommender systems

# Content-based Recommendation: An Example




User profile


Items recommended

# Content-based Recommendation: Algorithm

- We represent user profiles and item descriptions by vectorizing them using a set of $k$ keywords (e.g., using **TF-IDF**)

$$I_j = (i_{j,1}, i_{j,2}, \ldots, i_{j,k})$$
$$U_i = (u_{i,1}, u_{i,2}, \ldots, u_{i,k})$$

- Compute their (cosine) similarity

$$sim(U_i, I_j) = cos(U_i, I_j) = \frac{\sum_{l=1}^{k} u_{i,l} i_{j,l}}{\sqrt{\sum_{l=1}^{k} {u_{i,l}}^2} \sqrt{\sum_{l=1}^{k} {i_{j,l}}^2}}$$

- Recommend the top most similar items to the user

**Algorithm** Content-based recommendation

**Require:** User $i$'s Profile Information, Item descriptions for items $j \in \{1, 2, \ldots, n\}$, $k$ keywords, $r$ number of recommendations.

1: **return** $r$ recommended items.
2: $U_i = (u_1, u_2, \ldots, u_k)$ = user $i$'s profile vector;
3: $\{I_j\}_{j=1}^{n} = \{(i_{j,1}, i_{j,2}, \ldots, i_{j,k})$ = item $j$'s description vector$\}_{j=1}^{n}$;
4: $s_{i,j} = sim(U_i, I_j), 1 \leq j \leq n$;
5: Return top $r$ items with maximum similarity $s_{i,j}$.

# 2. Rating-based Recommendation

- AKA Collaborative filtering: recommend items by only **users' past behavior**
- Advantage: we don't need to have additional information about the users or content of the items
  - Users' **rating** or **purchase history** is the only information that is needed to work
- Input: Rating matrix
  - Users rate (rank) items (purchased, watched)
  - Explicit ratings: entered by a user directly
  - Implicit ratings: inferred from other user behavior
    - e.g. the amount of time users spent on a webpage
    - e.g. the number of times users listen to a song

| Movie | Alice (1) | Bob (2) | Carol (3) | Dave (4) |
|---|---|---|---|---|
| Love at last | 5 | 5 | 0 | 0 |
| Romance forever | 5 | ? | ? | 0 |
| Cute puppies of love | ? | 4 | 0 | ? |
| Nonstop car chases | 0 | 0 | 5 | 4 |
| Swords vs. karate | 0 | 0 | 5 | ? |

# User-based CF vs. Item-based CF

| | I1 | I2 | I3 | I4 |
|---|---|---|---|---|
| U1 | 1 | 2 | 4 | 4 |
| U2 | 1 | 2 | 4 | ? |
| U3 | 2 | 5 | 2 | 2 |
| U4 | 5 | 2 | 3 | 3 |

- Find similar users to me and recommend what they liked
- Why?: users with similar **previous** ratings for items are likely to rate future items similarly

| | I1 | I2 | I3 | I4 |
|---|---|---|---|---|
| U1 | 1 | 2 | 4 | 4 |
| U2 | 1 | 2 | 4 | ? |
| U3 | 2 | 5 | 2 | 2 |
| U4 | 5 | 2 | 3 | 3 |

- Find similar items to those that I have previously liked
- Why?: Items that have received similar ratings **previously** from users are likely to receive similar ratings from future users

# Collaborative Filtering: Algorithm

**User-based**

1. Weigh all users with respect to their similarity with the current user
2. Select a subset of the users (e.g. top-k neighbors) as recommenders
3. Predict the rating of the user for specific items using neighbors' ratings for the same (or similar) users
4. Recommend items with the highest predicted rank

**Item-based**

1. Weigh all items with respect to their similarity with the current item
2. Select a subset of the items (e.g. top-k neighbors) as recommenders
3. Predict the rating of the user for specific items using neighbors' ratings for the same (or similar) items
4. Recommend items with the highest predicted rank

# User-based vs. Item-based CF

- Finds the **most similar users** to the current user
- Cosine Similarity:

- Pearson Similarity (Correlation Coefficient):

- Finds the **most similar items** to the current item
- Cosine Similarity:

- Pearson Similarity (Correlation Coefficient):

# User-based vs. Item-based CF

- Update the ratings:

$r_{u,i} = r_u + \sum_{v \in N(u)} sim(u,v)(r_{v,i} - r_v) / \sum_{v \in N(u)} sim(u,v)$

$r_{u,i}$ : predicted rating of user $u$ for item $i$

$r_u$ : user u's mean rating

$r_v$ : user v's mean rating

$r_{v,i}$ : observed rating of user v for item i

- Update the ratings:

$r_{u,i} = r_i + \sum_{j \in N(i)} sim(i,j)(r_{u,j} - r_j) / \sum_{j \in N(i)} sim(i,j)$

$r_i$ : item i's mean rating

$r_j$ : item j's mean rating

# User-based CF, Example

| | Lion King | Aladdin | Mulan | Anastasia |
|---|---|---|---|---|
| John | 3 | 0 | 3 | 3 |
| Joe | 5 | 4 | 0 | 2 |
| Jill | 1 | 2 | 4 | 2 |
| Jane | 3 | ? | 1 | 0 |
| Jorge | 2 | 2 | 0 | 1 |

Predict Jane's rating for Aladdin

## 1- Calculate average ratings

$$\bar{r}_{John} = \frac{3+3+0+3}{4} = 2.25$$

$$\bar{r}_{Joe} = \frac{5+4+0+2}{4} = 2.75$$

$$\bar{r}_{Jill} = \frac{1+2+4+2}{4} = 2.25$$

$$\bar{r}_{Jane} = \frac{3+1+0}{3} = 1.33$$

$$\bar{r}_{Jorge} = \frac{2+2+0+1}{4} = 1.25$$

## 2- Calculate user-user similarity

$$sim(Jane, John) = \frac{3\times3+1\times3+0\times3}{\sqrt{10}\sqrt{27}} = 0.73$$

$$sim(Jane, Joe) = \frac{3\times5+1\times0+0\times2}{\sqrt{10}\sqrt{29}} = 0.88$$

$$sim(Jane, Jill) = \frac{3\times1+1\times4+0\times2}{\sqrt{10}\sqrt{21}} = 0.48$$

$$sim(Jane, Jorge) = \frac{3\times2+1\times0+0\times1}{\sqrt{10}\sqrt{5}} = 0.84$$

# User-based CF, Example (cont'd)

3- Calculate Jane's rating for Aladdin, Assume that neighborhood size = 2

$$r_{Jane,Aladdin} = \bar{r}_{Jane} + \frac{sim(Jane,Joe)(r_{Joe,Aladdin} - \bar{r}_{Joe})}{sim(Jane,Joe) + sim(Jane,Jorge)}$$

$$+ \frac{sim(Jane,Jorge)(r_{Jorge,Aladdin} - \bar{r}_{Jorge})}{sim(Jane,Joe) + sim(Jane,Jorge)}$$

$$= 1.33 + \frac{0.88(4 - 2.75) + 0.84(2 - 1.25)}{0.88 + 0.84} = 2.33$$

# Item-based CF, Example

## 1- Calculate average ratings

$$\bar{r}_{Lion\ King} = \frac{3+5+1+3+2}{5} = 2.8$$

$$\bar{r}_{Aladdin} = \frac{0+4+2+2}{4} = 2.$$

$$\bar{r}_{Mulan} = \frac{3+0+4+1+0}{5} = 1.6$$

$$\bar{r}_{Anastasia} = \frac{3+2+2+0+1}{5} = 1.6$$

## 2- Calculate item-item similarity

$$sim(Aladdin, Lion\ King) = \frac{0\times3+4\times5+2\times1+2\times2}{\sqrt{24}\sqrt{39}} = 0.84$$

$$sim(Aladdin, Mulan) = \frac{0\times3+4\times0+2\times4+2\times0}{\sqrt{24}\sqrt{25}} = 0.32$$

$$sim(Aladdin, Anastasia) = \frac{0\times3+4\times2+2\times2+2\times1}{\sqrt{24}\sqrt{18}} = 0.67$$

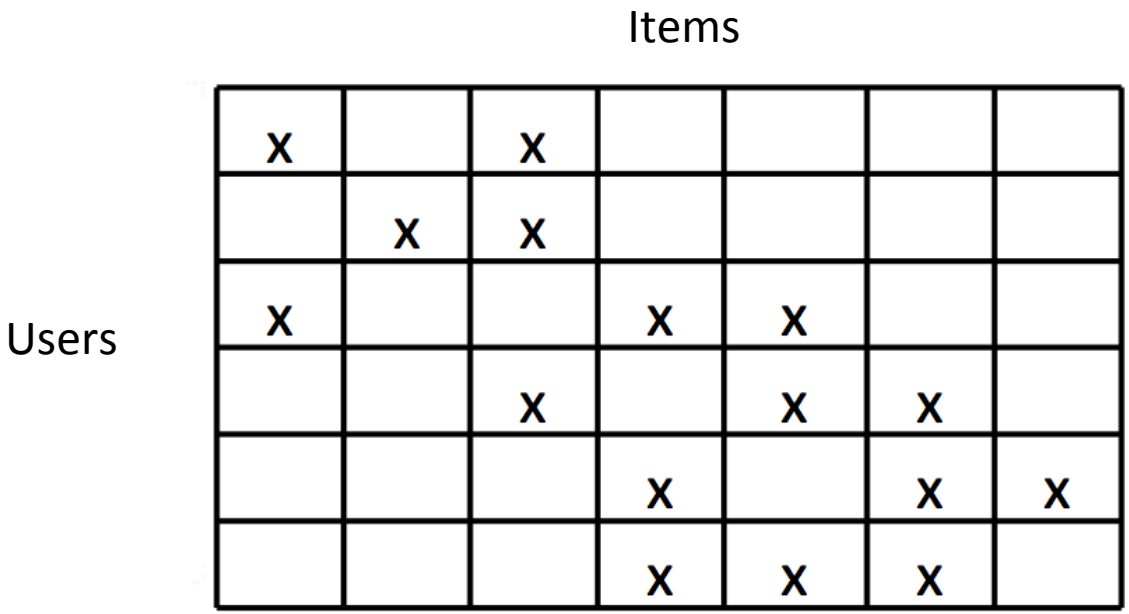
## 3- Calculate Jane's rating for Aladdin, Assume that neighborhood size = 2

$$r_{Jane,Aladdin} = \bar{r}_{Aladdin} + \frac{sim(Aladdin, Lion\ King)(r_{Jane,Lion\ King} - \bar{r}_{Lion\ King})}{sim(Aladdin, Lion\ King) + sim(Aladdin, Anastasia)} + \frac{sim(Aladdin, Anastasia)(r_{Jane,Anastasia} - \bar{r}_{Anastasia})}{sim(Aladdin, Lion\ King) + sim(Aladdin, Anastasia)}$$

$$= 2 + \frac{0.84(3-2.8) + 0.67(0-1.6)}{0.84+0.67} = 1.40$$

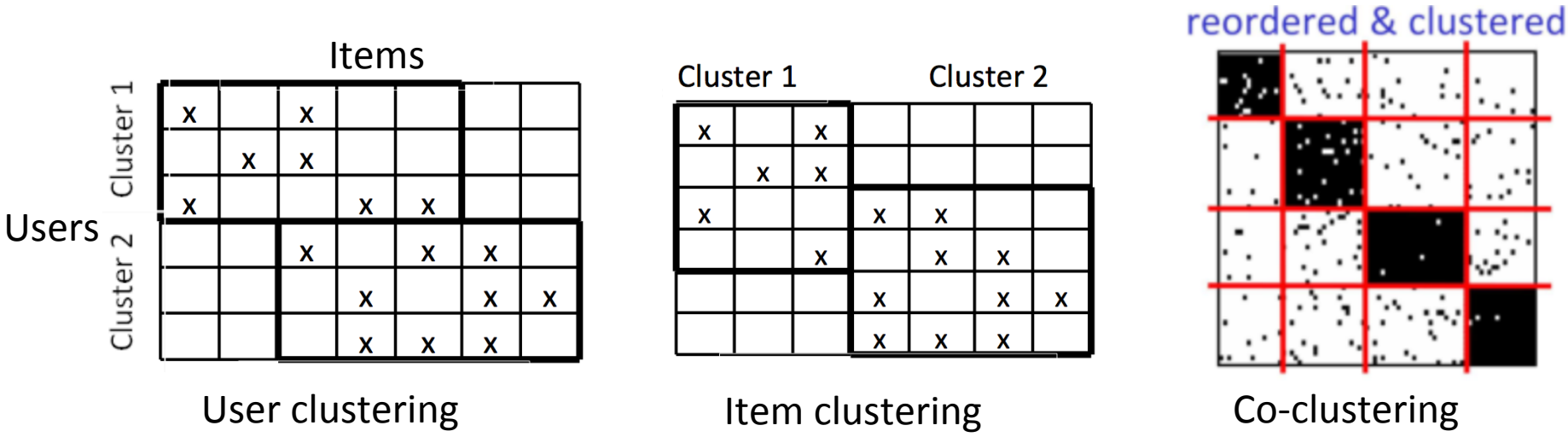
# 3. Clustering-based Recommendation

- **Limitations** of rating-based recommendation:
  - Hard to **scale** with large data
  - Bad with **sparse** rating matrix
  - Bad with **diversity** of users and items



# Clustering-based Recommendation

- Solution: **clustering** the data before-hand
  - Clustering based on ratings: k-means [*], etc.
  - **One-dimension clustering**: user clustering, item clustering
  - **Co-clustering**
- Then, perform rating-based recommendation on each cluster




[*] Al Mamunur Rashid, Shyong K. Lam, George Karypis, and John Riedl. "ClustKNN: a highly scalable hybrid model-& memory-based CF algorithm." *Proceeding of WebKDD* 2006 (2006).

# Clustering-based Recommendation – How does it work?

- Any customer that shall be classified as a member of **CLUSTER** will received recommendations based on preferences of the group:
  - Book 2 will be highly recommended to Customer F
  - Book 6 will also be recommended to some extent
- Pros:
  - Overcome the sparse data problem
  - Capture latent similarities between users and items
- Cons:
  - Recommendations (per cluster) maybe less relevant than collaborative filtering (per individual)

| | Book1 | Book2 | Book3 | Book4 | Book5 | Book6 |
|---|---|---|---|---|---|---|
| Customer A | X | | | X | | |
| Customer B | | X | X | | X | |
| Customer C | | X | X | | | |
| Customer D | | X | | | | X |
| Customer E | X | | | | X | |
| Customer F | | | X | | X | |

# RecSys - Performance Measures

- Qualitative measures:
  - **User Satisfaction** (e.g. questionnaire)
- Quantitative measures:
  - If ground truth is not available:
    - Add-on sales
    - Click-through rates
    - Number of products purchased
  - If ground truth is available:
    - **Predictive accuracy**:
      - The ratio of predicted ratings being the true user ratings?
    - Rank accuracy

# Predictive accuracy

- **Mean Absolute Error (*MAE*).** The average absolute deviation between a predicted rating ($p$) and the user's true rating ($r$)
  - $NMAE = MAE/(rmax - rmin)$

$$MAE = \frac{\sum_{ij} |\hat{r}_{ij} - r_{ij}|}{n}$$

- **Root Mean Square Error (*RMSE*).** Similar to *MAE*, but places more emphasis on larger deviation

$$RMSE = \sqrt{\frac{1}{n} \sum_{i,j} (\hat{r}_{ij} - r_{ij})^2}$$

# Evaluation Example

| *Item* | *Predicted Rating* | *True Rating* |
|---|---|---|
| *1* | *1* | *3* |
| *2* | *2* | *5* |
| *3* | *3* | *3* |
| *4* | *4* | *2* |
| *5* | *4* | *1* |

$$MAE = \frac{|1-3| + |2-5| + |3-3| + |4-2| + |4-1|}{5} = 2$$

$$NMAE = \frac{MAE}{5-1} = 0.5$$

$$RMSE = \sqrt{\frac{(1-3)^2 + (2-5)^2 + (3-3)^2 + (4-2)^2 + (4-1)^2}{5}}$$

$$= 2.28$$

# Rank Accuracy

- **Kendall's $\tau$**
  - **Compares concordant** the items of the recommended ranking list **against the ground truth** ranking list
    - If the two orders are consistent, it is concordant
    - E.g., for top 4 items in ranking list, there are 4×3/2=6 pairs

$$\tau = c - d / (n \choose 2)$$

    - $c$ is the number of concordants
    - $d$ is the number of disconcordants

# Ranking Accuracy: Example

- Consider a set of four items $I = \{i1, i2, i3, i4\}$ for which the predicted and true rankings are as follows

| | *Predicted Rank* | *True Rank* |
|---|---|---|
| $i_1$ | 1 | 1 |
| $i_2$ | 2 | 4 |
| $i_3$ | 3 | 2 |
| $i_4$ | 4 | 3 |

Pair of items and their status {**concordant**/**discordant**} are

$(i_1, i_2)$ : *concordant*

$(i_1, i_3)$ : *concordant*

$(i_1, i_4)$ : *concordant*

$(i_2, i_3)$ : *discordant*

$(i_2, i_4)$ : *discordant*

$(i_3, i_4)$ : *concordant*

$$\tau = \frac{4 - 2}{6} = 0.33$$

# References


[1] https://blog.exploratory.io/find-correlation-or-similarity-among-categories-or-variables-4813130f53c0

[2] https://blog.exploratory.io/an-introduction-to-regression-analysis-in-exploratory-9422237c0ff8

[3] https://www.slideshare.net/PhamCuong/clustering-technique-for-collaborative-filtering-recommendation-and-application-to-venue-recommendation

[4] R. Zafarani, M. A. Abbasi, and H. Liu, Social Media Mining: An Introduction, Cambridge University Press, 2014. http://socialmediamining.info/

[5] https://www.slideshare.net/xamat/recommender-systems-machine-learning-summer-school-2014-cmu

[6] Business Forecasting: Pearson New International Edition, 9/E Hanke & Wichern ©2014 | Pearson | Published: 15 Aug 2013 ISBN-10: 1292023007 | ISBN-13: 9781292023007

[7] http://business.unr.edu/faculty/ronlembke/handouts/Seasonality%20Final17.pdf