Topic 2: Collaborative Filtering

(15 total points) An online gaming platform has implemented a user-based recommender system with implicit feedback elicitation. Users are not requested to vote for their preferred game or express their likes and dislikes explicitly (i.e. a five-point scale), as the platform considers that its users can manipulate their responses in a way to bias the recommender system towards recommending a specific game. The platform takes user behavior into account instead, that is considered more adequate, reflects the actual user likes and is less prone to attacks (i.e. fake accounts and fake likes/dislikes). The approach is to record the time a user spends playing a game on the platform per month and use this as a score of user like/dislike for the game. A user profile is built for each game, and this is used to serve recommendations to other/new users.

Following the above, assume that the table below holds the time (in seconds) that the five most active users spent playing the five most popular console games on the platform.

User	Console Game Title	Time (sec)
User 1	Call of Duty	45871
User 1	NBA	103554
User 1	Halo	12
User 1	Dragon Ball	213214
User 1	Super Mario	1568
User 2	Call of Duty	4
User 2	Dragon Ball	232581
User 2	NBA	985
User 2	Halo	974
User 3	NBA	112548
User 3	Call of Duty	9854
User 3	Super Mario	54
User 3	Halo	412
User 4	NBA	32145
User 4	Super Mario	15687
User 4	Halo	23587
User 4	Call of Duty	15420
User 4	Dragon Ball	458
User 5	Call of Duty	29214
User 5	Halo	18987
User 5	Dragon Ball	16523
User 5	Super Mario	685
User 5	NBA	8654

Answer:

	Super Mario	NBA	Dragon Ball	Call of Duty	Halo
User 1	4	6	6	5	2
User 2	N/A	3	6	1	3
User 3	2	6	N/A	4	3
User 4	5	5	3	5	5
User 5	3	4	5	5	5

(a) (2 points) Convert the time data into ratings using the following formula.

$$rating_{user,game} = \lceil log_{10}(time\ in\ seconds_{user,game}) \rceil$$

where the $\lceil . \rceil$ function is the ceiling function, defined as "a function that maps a real number x to the least integer greater than or equal to x". E.g. $\lceil 2{,}54 \rceil = 3, \lceil 2{,}001 \rceil = 3$.

Fill in the missing values in the utility matrix below noted with placeholders (?).

Answer:

	Super Mario	NBA	Dragon Ball	Call of Duty	Halo
User 1	?	6	?	5	?
User 2		?	6	?	?
User 3	2	6		?	3
User 4	?	?	?	5	?
User 5	?	?	5	?	5

(a) $rating_{1,SM} = \log(1568) \simeq 3{,}195 \quad \lceil 3{,}195 \rceil = 4$

$rating_{2,NBA} = \log(103554) \simeq 5{,}015 \quad \lceil 5{,}015 \rceil = 6$

$rating_{1,DB} = \log(213214) \simeq 5{,}329 \quad \lceil 5{,}329 \rceil = 6$

$rating_{1,CoD} = \log(45871) \simeq 4{,}661 \quad \lceil 4{,}661 \rceil = 5$

$rating_{1,H} = \log(12) \simeq 1{,}079 \quad \lceil 1{,}079 \rceil = 2$

Similarly for the other users.

(b) (4 points) Assume that the platform wants to make recommendations to a user with the following gaming history in the platform, using the profiles of the five most active users (as calculated in answer 2.a).

User	Console Game Title	Time (sec)
User 6	NBA	68547
User 6	Call of Duty	984
User 6	Super Mario	3254

convert raw times →

$rating_{6,NBA} = \log(68547) \simeq 4{,}836 \quad \lceil 4{,}836 \rceil = 5$

$rating_{6,CoD} = \log(984) \simeq 2{,}993 \quad \lceil 2{,}993 \rceil = 3$

$rating_{6,SM} = \log(3254) \simeq 3{,}512 \quad \lceil 3{,}512 \rceil = 4$

Following that the platform recommender system is a user based one, calculate the user-user similarity pairs shown in the box below, using the Pearson Correlation Coefficient as the similarity measure. Do all calculations with the greatest accuracy. Report the final results with 4 decimal places of accuracy (do not round up or down the numbers at any calculation step).

Fill in the missing values noted with placeholders (?) in the user/user table below with the correct values.

ANSWER

	User 6
User 1	?
User 2	0,624695
User 3	?
User 4	?
User 5	?

	Super Mario	NBA	Dragon Ball	Call of Duty	Halo
User 1	4	6	6	5	2
User 2	—	3	6	1	3
User 3	2	6	—	4	3
User 4	5	5	3	5	5
User 5	3	4	5	5	5
User 6	4	5	—	3	—

We consider only the games both have rated

User	Super Mario	NBA	CoD
1	4	6	5
2	—	3	1
3	2	6	4
4	5	5	5
5	3	4	5
6	4	5	3

We will use the Pearson Correlation Coefficient

$$sim(u,v) = \frac{\sum_{\forall i \in S} (r_{u,i} - \bar{r}_u) \cdot (r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{\forall i \in S} (r_{u,i} - \bar{r}_u)^2} \cdot \sqrt{\sum_{\forall i \in S} (r_{v,i} - \bar{r}_v)^2}}$$

where $S = I_u \cap I_v$ is the total number of items rated in common by users u and v.

$r_{u,i}$ and r_{vi}: the ratings of users u and v for item i

$\bar{r}_u$ and $\bar{r}_v$: the mean ratings of users u and v

We calculate the means:

$\bar{r}_1 = \frac{4+6+6+5+2}{5} = \frac{23}{5} = 4{,}6$

$\bar{r}_2 = \frac{3+6+1+3}{4} = \frac{13}{4} = 3.25$

$\bar{r}_3 = \frac{2+6+4+3}{4} = \frac{15}{4} = 3.75$

$\bar{r}_4 = \frac{5+5+3+5+5}{5} = \frac{23}{5} = 4{,}6$

$\bar{r}_5 = \frac{3+4+5+5+5}{5} = 4.4$

$\bar{r}_6 = \frac{4+5+3}{3} = 4$

$$sim(1,6) = \frac{(r_{1,SM} - \bar{r}_1)\cdot(r_{6,SM} - \bar{r}_6) + (r_{1,NBA} - \bar{r}_1)(r_{6,NBA} - \bar{r}_6) + (r_{1,COD} - \bar{r}_1)(r_{6,COD} - \bar{r}_6)}{\sqrt{(r_{1,SM} - \bar{r}_1)^2 + (r_{1,NBA} - \bar{r}_1)^2 + (r_{1,COD} - \bar{r}_1)^2} \cdot \sqrt{(r_{6,SM} - \bar{r}_6)^2 + (r_{6,NBA} - \bar{r}_6)^2 + (r_{6,COD} - \bar{r}_6)^2}} =$$

$$= \frac{(4-4{,}6)\cdot(4-4) + (6-4{,}6)\cdot(5-4) + (5-4{,}6)(3-4)}{\sqrt{(4-4.6)^2 + (6-4.6)^2 + (5-4.6)^2}\sqrt{(4-4)^2 + (5-4)^2 + (3-4)^2}} = \frac{1{,}4 - 0{,}4}{\sqrt{0{,}36 + 1{,}96 + 0{,}16}\cdot\sqrt{2}} = \frac{1}{\sqrt{4{,}96}} = 0{,}4490$$

Similarly

$sim(2,6) = 0{,}624695$ max #1

$sim(3,6) = 0.4942$ max #2

$sim(4,6) = 0.0000$

$sim(5,6) = -0.4490$

(c) (1 point) Assuming that the neighborhood size is 2 (and the Pearson Correlation coefficient as the similarity measure), indicate User 6 neighbors below. Fill in the missing values noted with placeholders (?) with the id of the users.

Answer:

User 6 neighbors: User 4, User 3

(c) (3 points) Use the Adjusted Weighted Average Rating (formula 9.5 in section 9.2.2 of "*R. Zafarani, M.A. Abbasi & H. Liu (2014). Social media mining: an introduction. Cambridge University Press*") to predict User 6 ratings for games not played and the recommendation priority, assuming that the neighborhood size is 2 and the recommender system is user-based.
Do not normalize ratings before calculating the similarities. Report the final result with 4 decimal digits of accuracy (do not round up or down the number). Fill in the missing values noted with placeholders (?).

Answer:

(1 point) User 6/Dragon Ball:?

(1 point) User 6/Halo:?

(1 point) Recommended games: First: ?, Second: ?

User-Based Collaborative Filtering. In this method, we predict the rating of user u for item i by (1) finding users most similar to u and (2) using a combination of the ratings of these users for item i as the predicted rating of user u for item i. To remove noise and reduce computation, we often limit the number of similar users to some fixed number. These most similar users are called the *neighborhood* for user u, $N(u)$. In user-based collaborative filtering, the rating of user u for item i is calculated as

$$r_{u,i} = \bar{r}_u + \frac{\sum_{v \in N(u)} sim(u,v)(r_{v,i} - \bar{r}_v)}{\sum_{v \in N(u)} sim(u,v)}, \quad (9.5)$$

where the number of members of $N(u)$ is predetermined (e.g., top 10 most similar members).

- $r_{6,DB} = \bar{r}_6 + \frac{\sum_{i=2}^{3} sim(6,i)\cdot(r_{i,DB} - \bar{r}_i)}{\sum_{i=2}^{3} sim(6,i)} =$

did not rate DB

$$= 4 + \frac{sim(6,2)(r_{2,DB} - \bar{r}_2) + sim(6,3)(r_{3,DB} - \bar{r}_3)}{sim(6,2) + sim(6,3)} = 4 + \frac{0.624695\ (6 - 3.25)}{0{,}624695 + 0{,}4942} = 4 + \frac{1.7179125}{1.118895} = 5{,}5353$$

- $r_{6,Halo} = \bar{r}_6 + \frac{sim(6,2)\cdot(r_{2,Halo} - \bar{r}_2) + sim(6,3)\cdot(r_{3,Halo} - \bar{r}_3)}{sim(6,2) + sim(6,3)} = 4 + \frac{0.624695\cdot(3-3.25) + 0.4942\cdot(3-3.75)}{1.118895} = 3.5291$

	Super Mario	NBA	Dragon Ball	Call of Duty	Halo
USER 6	4	5	5,5353	3	3,5291

Recommended Games: #1 Dragon Ball #2 Halo

(d) (5 points) Assume that a malicious user wants to manipulate platform recommendations and bias towards a specific game. The user does not know details of the recommender system algorithm but being a user of the platform and receiving recommendations, intuitively assumes that rating is based on playing duration and popular games have an overall high presence in recommendations. To attack the system, the malicious user follows a "Bandwagon Attack" scheme. The bandwagon attack intuition is that ratings (duration of play) of popular items are high in many users. So, to increase the chances for a fake user to become a member of the neighbors of some user, the popular item should be set to the maximum possible rating, while the other items will be rated low in the fake account. This way, as a neighbor, the fake user will bias the recommendation towards a specific item. To achieve this, the malicious user builds two fake users and spends some time playing in order to build the appropriate user profiles. The table below depicts the data for the two fake users.

convert raw times

User	Console Game Title	Time (sec)	rating (log[time])
Fake 1	Super Mario	5	1
Fake 1	NBA	16250	5
Fake 1	Dragon Ball	56	2
Fake 1	Call of Duty	3	1
Fake 1	Halo	2	1
Fake 2	Super Mario	2	1
Fake 2	NBA	15220	5
Fake 2	Dragon Ball	3	1
Fake 2	Call of Duty	4	1
Fake 2	Halo	20	2

Considering the two Fake users profiles examine whether the attack is successful or not. (i.e. to change the order of recommendations). To do this, find the new neighbors, predict again User 6 ratings for games not played and the recommendation priority, assuming as previously that the neighborhood size is 2 and the recommender system is user-based.

Do not normalize ratings before calculating the similarities. Report the final result with 4 decimal digits of accuracy (do not round up or down the number). Fill in the missing values noted with placeholders (?).

	Super Mario	NBA	Dragon Ball	Call of Duty	Halo
User 1	4	6	6	5	2
User 2	—	3	6	1	3
User 3	2	6	—	4	3
User 4	5	5	3	5	5
User 5	3	4	5	5	5
User 6	4	5	—	3	—
Fake 1	1	5	2	1	1
Fake 2	1	5	1	1	2

$\overline{r}_{F_1} = \frac{1+5+2+1+1}{5} = 2$

$\overline{r}_{F_2} = \frac{1+5+1+1+2}{5} = 2$

$sim(F_1,6) = 0,8528$ max #1

$sim(F_2,6) = 0,8528$ max #2

new user 6 neighbours: Fake 1, Fake 2

Predict the user Rating again

- $r_{6,DB} = \overline{r_6} + \frac{\sum_{i=F_1}^{F_2} sim(6,i)\cdot(r_{i,DB} - \overline{r_i})}{\sum_{i=F_1}^{F_2} sim(6,i)} =$

$= 4 + \frac{0,8528\cdot(2-2) + 0,8528\cdot(1-2)}{2\cdot 0,8528} = 4 + \frac{-0,8528}{2\cdot 0,8528} =$

$= 4 - \frac{1}{2} = 3.5$

- $r_{6,HALO} = 4 + \frac{0,8528\cdot(1-2) + 0,8528\cdot(2-2)}{2\cdot 0,8528} = 4 - \frac{1}{2} = 3.5$

	Super Mario	NBA	Dragon Ball	Call of Duty	Halo
USER 6	4	5	3.5	3	3.5

Recommended New Games

#1 Dragon Ball

#2 Halo

Answer:

(2 points)

User 6 neighbors: Fake 1, Fake 2

(2 points)

User 6/Dragon Ball: 3.5

User 6/Halo: 3.5

(1 point)

Recommended games: #1 DragonBall #2 Halo **/ Attack was successful (YES/NO):** NO

Topic 3: Recommender System Evaluation

(15 total points) Answer the question below.

(a) (6 points) Assume the following utility matrix shows actual user ratings of 1–5 for five books. Missing cells indicate the user has not rated that item. We have three users (John, Mary, Sam) and five items: 1) War and Peace, 2) Anna Karenina, 3) Pride and Prejudice, 4)The Great Gatsby, and, 5)To Kill a Mockingbird

User	War and Peace	Anna Karenina	Pride and Prejudice	The Great Gatsby	To Kill a Mockingbird
John	4.5			5	5
Mary		5	4	4	
Sam	3		4.5		4

For the same users and items, an imaginary recommendation system has predicted the following ratings:

User	War and Peace	Anna Karenina	Pride and Prejudice	The Great Gatsby	To Kill a Mockingbird
John	4	4	3	**4**	**5**
Mary	**5**	5	2	**3**	3
Sam	**3**	5	**4**	2	**4**

i. Calculate the Mean Absolute Error (MAE) for the predicted ratings[1]:

ii. Calculate the Root Mean Squared Error (RMSE) for the predicted ratings[1]:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i,j}(\hat{r}_{ij} - r_{ij})^2} = \sqrt{\frac{1}{9}\left[(0,5)^2 + 1^2 + 2^2 + 1^2 + (0,5)^2\right]} =$$

$$= \frac{\sqrt{0.25 + 1 + 4 + 1 + 0.25}}{3} = \frac{\sqrt{6.5}}{3} = 0,84$$

(b) (9 points) A recommendation system outputs a top-5 ranking of music albums for a particular user. For each recommended album, we have a relevance score from the user on a 0–5 scale, where 5 is "most relevant." Suppose we define an album as "liked" if its relevance is 4 or 5. (Any score < 4 means the user does not count it as liked.)

Fill all the cells containing the question marks ?, as simplified ratios with the correct values in columns Precision@N, Recall@N and nDCGpos. Report the result with 2 decimal point accuracy. Do not round up or down the calculated value.

Rank	Album	Relevance	Precision@N	Recall@N	NDCGpos
1	The Beatles – Sgt. Pepper's Lonely Hearts Club Band	3 ✗	$\frac{0}{1} = 0$	$\frac{0}{3} = 0$	$\frac{3}{5} = 0.6$
2	Pink Floyd – The Dark Side of the Moon	5 ✓	$\frac{1}{2} = 0,5$	$\frac{1}{3} = 0,33$	$\frac{6.15}{8,15}$ 6.15 $= 0.75$
3	Michael Jackson – Thriller	4 ✓	$\frac{2}{3} = 0,66$	$\frac{2}{3} = 0.66$	$\frac{8.15}{10.15} = 0.80$
4	Fleetwood Mac – Rumours	2 ✗	$\frac{2}{4} = 0.50$	$\frac{2}{3} = 0.66$	$\frac{9.01}{11.44} = 0.78$
5	Nirvana – Nevermind	5 ✓	$\frac{3}{5} = 0.60$	$\frac{3}{3}$ 1.00	$\frac{10.95}{12.22} = 0.89$

(a) $MAE = \frac{\sum |\hat{r}_{ij} - r_{ij}|}{n}$

where n is the number of predicted ratings, $\hat{r}_{ij}$ is the predicted rating, and r_{ij} is the true rating.

User	War and Peace	Anna Karenina	Pride and Prejudice	The Great Gatsby	To Kill a Mockingbird
John	4.5, 4	N/A, 4	N/A, 3	5, 4	5, 5
Mary	N/A, 5	5, 5	4, 2	4, 3	N/A, 3
Sam	3, 3	N/A, 5	4.5, 4	N/A, 2	4, 4

John	0,5	–	–	1	0
Mary		0	2	1	–
Sam	0	–	0,5	–	0

Sum of absolute error: $0,5 + 1 + 2 + 1 + 0,5 = 5$

Total number of ratings used: $N = 9$

$MAE = \frac{5}{9} = 0.55$

Precision @N = How many of the first N albums are "liked".

Recall @N = How many of the total liked albums were found in the first N recommendations

$$DCG_{pos} = \sum_{i=1}^{pos} \frac{relevance_i}{log_2(i+1)} \qquad IDCG_{pos} = \sum_{i=1}^{pos} \frac{relevance_i}{log_2(i+1)}$$

$$nDCG_{pos} = \frac{DCG_{pos}}{IDCG_{pos}}$$

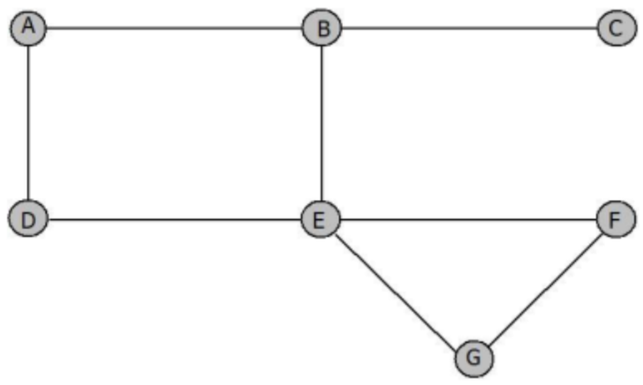
Rank	Relevance	DCG
1	3	$3/log_2 2 = 3/1 = 3$
2	5	$3 + \frac{5}{log_2 3} = 6.15$
3	4	$6.15 + \frac{4}{log_2 4} = 6.15 + \frac{4}{2} = 8.15$
4	2	$8.15 + \frac{2}{log_2 5} = 9.01$
5	5	$9.01 + \frac{5}{log_2 6} = 10.95$

Ideal Rank	Relevance	IDCG
1	5	$5/log_2(2) = 5/1 = 5$
2	5	$5 + \frac{5}{log_2(3)} = 8.15$
3	4	$8.15 + \frac{4}{log_2(4)} = 10.15$
4	3	$10.15 + \frac{3}{log_2(5)} = 11.44$
5	2	$11.44 + \frac{2}{log_2(6)} = 12.22$

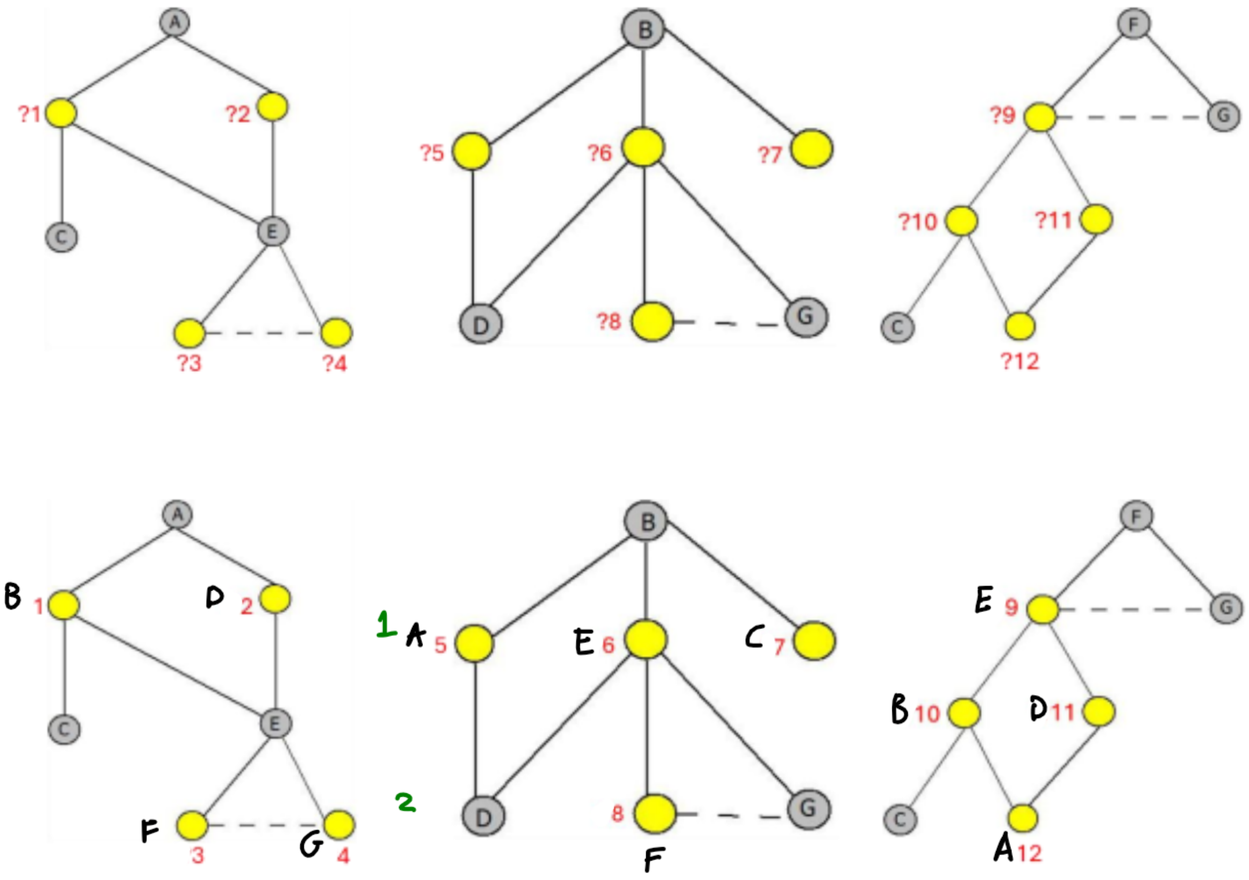
Topic 4: Girvan-Newman Algorithm

(15 total points) Answer the questions below.

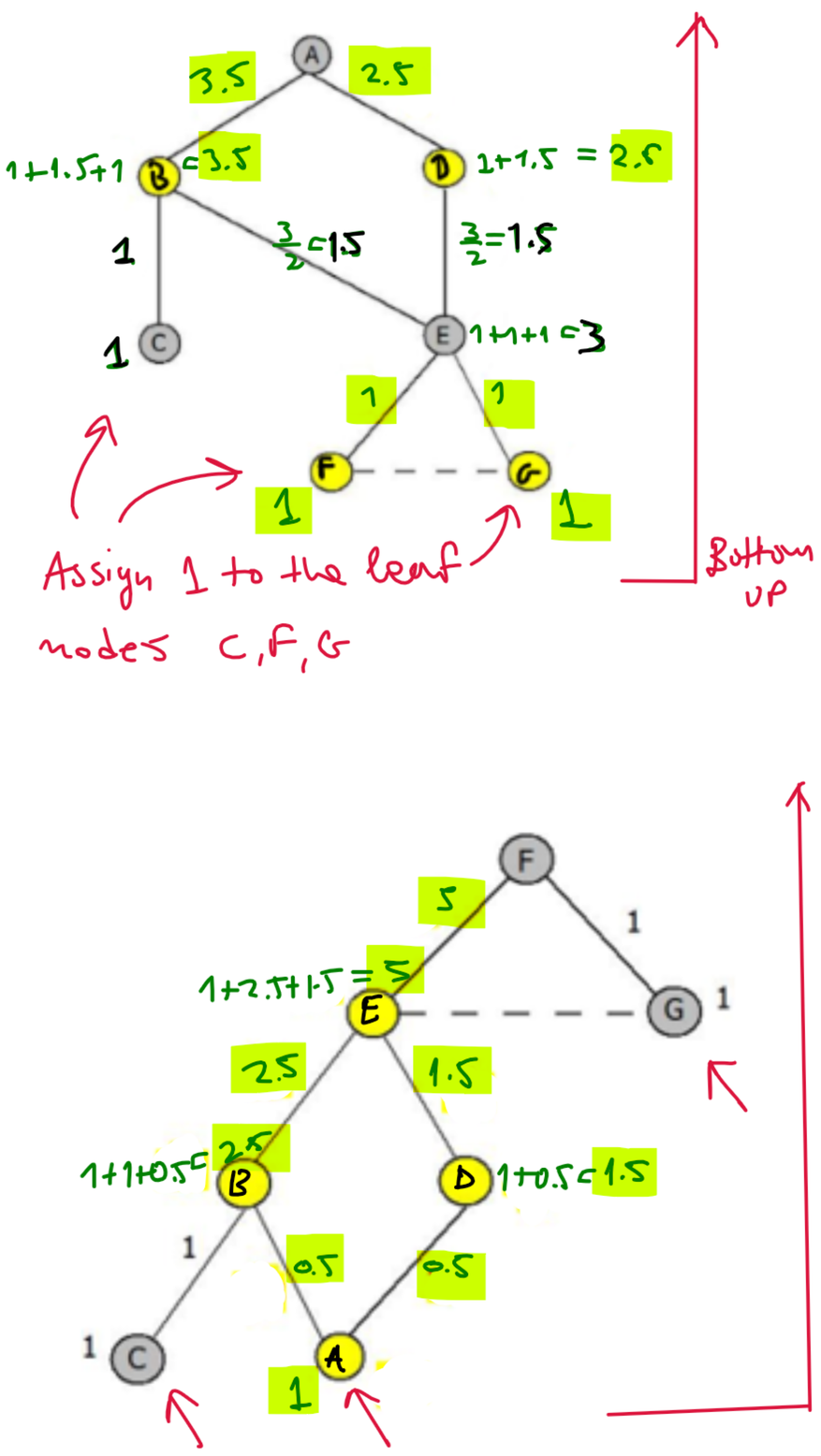
(a) (5 points) Suppose we have the following graph

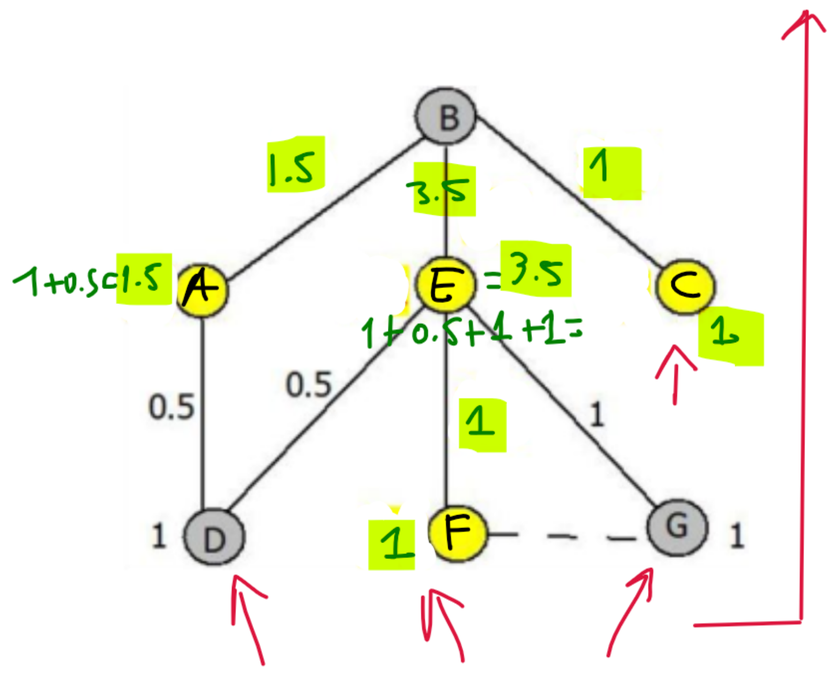


We would like to apply Breadth First Search (BFS) on the above graph three times with root nodes A, B and F respectively. In the following graphs fill in the missing nodes (denoted by yellow nodes and identified by numbered ?) with the correct node labels in order to apply correctly the BFS algorithm.



(b) (6 points) In the three graphs shown below, fill in the missing labels next to the nodes with the number of shortest paths that reach it from the root and the edge flows denoted by a numbered ? and yellow background according to the second and third step of the Girvan-Newman algorithm in order to find the betweenness of the edges (Note: label both nodes and edges bottom-up)





root A

1 = 3.5
2 = 2.5
3 = 3.5
4 = 2.5
5 = 1
6 = 1
7 = 1
8 = 1

root F

17 = 5
18 = 5
19 = 2.5
20 = 1.5
21 = 2.5
22 = 1.5
23 = 0.5
24 = 0.5
25 = 1

root B

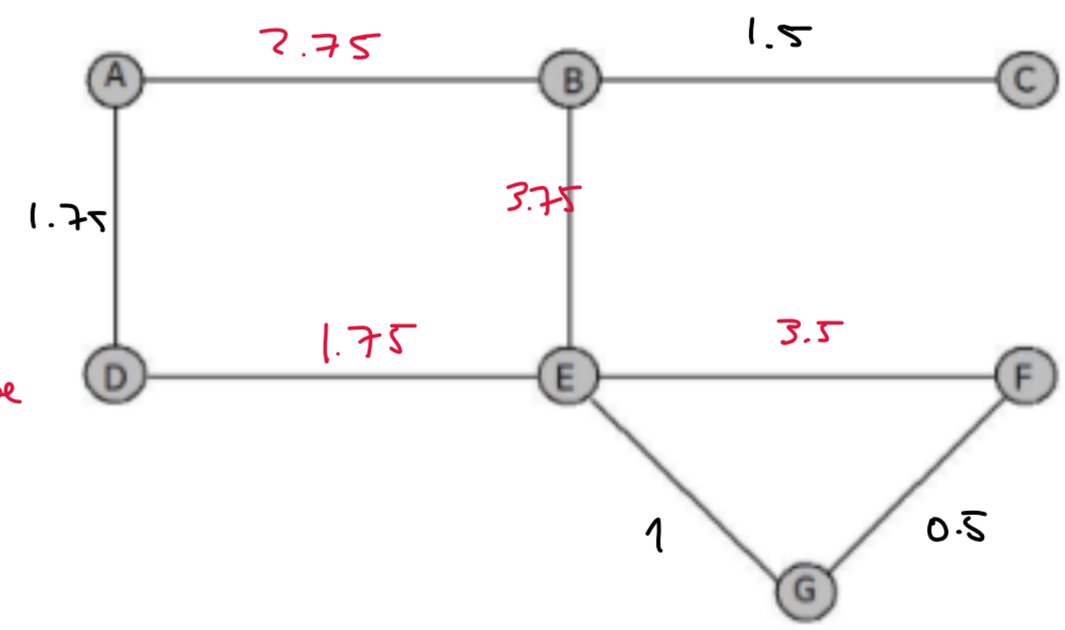
9 = 1.5
10 = 1
11 = 1.5
12 = 3.5
13 = 3.5
14 = 1
15 = 1
16 = 1

(c) (4 points) Answer the following two questions.

i) In order to apply the Girvan – Newman algorithm we have to perform a breadth-first search from each node of the graph taken as a root node. Suppose that for simplicity we take only the 3 previously used nodes as root nodes, meaning nodes A, B and F. Taking into account this assumption and the values calculated in the previous question, fill in the following graph the final betweenness values of Girvan-Newman Algorithm that are missing (denoted by a numbered ? and yellow background)

Edges	A	B	F	Total
AB	3.5	1.5	0.5	5.5
AD	2.5	0.5	0.5	3.5
BC	1	1	1	3
BE	3.5	1.5	2.5	7.5
DE	1.5	0.5	1.5	3.5
EF	1	1	5	7
EG	1	1		2
FG			1	1

divide all by 2 since all shortest paths will be calculated twice.



ii) Which is the edge with the highest betweenness centrality?

Answer:

The highest betweenness centrality has the edge {B,E} with value equal to 3.75