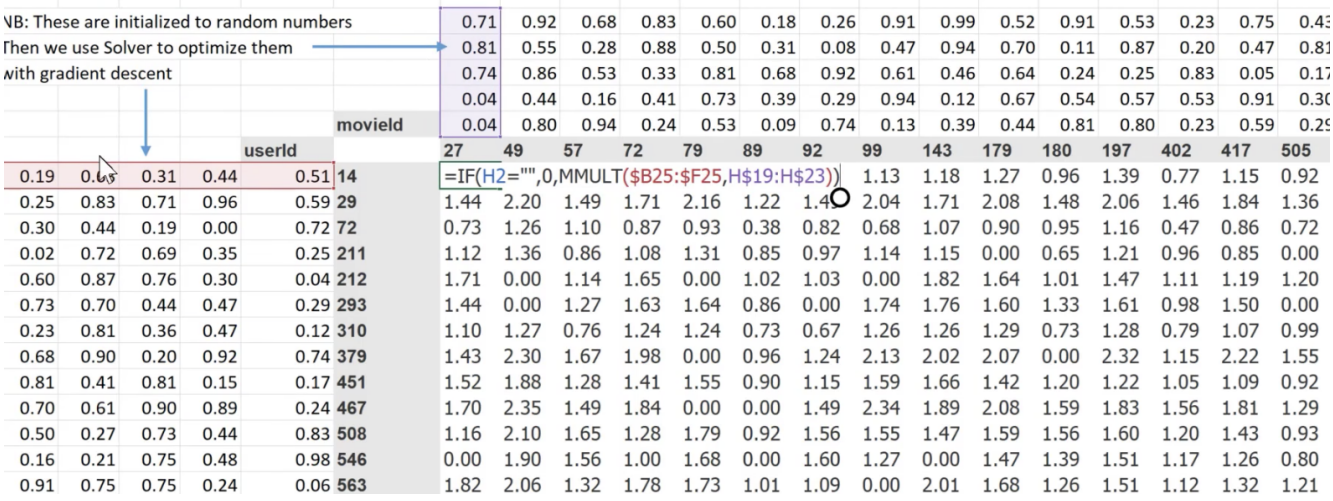
collaborative filtering

- One approach for recommendations is to naively take a dot product between movie embedding and user embedding, for example.



						0.71	0.92	0.68	0.83	0.60	0.18	0.26	0.91	0.99	0.52	0.91	0.53	0.23	0.75	0.43
						0.81	0.55	0.28	0.88	0.50	0.31	0.08	0.47	0.94	0.70	0.11	0.87	0.20	0.47	0.81
						0.74	0.86	0.53	0.33	0.81	0.68	0.92	0.61	0.46	0.64	0.24	0.25	0.83	0.05	0.17
						0.04	0.44	0.16	0.41	0.73	0.39	0.29	0.94	0.12	0.67	0.54	0.57	0.53	0.91	0.30
					movieId	0.04	0.80	0.94	0.24	0.53	0.09	0.74	0.13	0.39	0.44	0.81	0.80	0.23	0.59	0.29
				userId		27	49	57	72	79	89	92	99	143	179	180	197	402	417	505
0.19	0.65	0.31	0.44	0.51	14	=IF(H2="",0,MMULT($B25:$F25,H$19:H$23))							1.13	1.18	1.27	0.96	1.39	0.77	1.15	0.92
0.25	0.83	0.71	0.96	0.59	29	1.44	2.20	1.49	1.71	2.16	1.22	1.40	2.04	1.71	2.08	1.48	2.06	1.46	1.84	1.36
0.30	0.44	0.19	0.00	0.72	72	0.73	1.26	1.10	0.87	0.93	0.38	0.82	0.68	1.07	0.90	0.95	1.16	0.47	0.86	0.72
0.02	0.72	0.69	0.35	0.25	211	1.12	1.36	0.86	1.08	1.31	0.85	0.97	1.14	1.15	0.00	0.65	1.21	0.96	0.85	0.00
0.60	0.87	0.76	0.30	0.04	212	1.71	0.00	1.14	1.65	0.00	1.02	1.03	0.00	1.82	1.64	1.01	1.47	1.11	1.19	1.20
0.73	0.70	0.44	0.47	0.29	293	1.44	0.00	1.27	1.63	1.64	0.86	0.00	1.74	1.76	1.60	1.33	1.61	0.98	1.50	0.00
0.23	0.81	0.36	0.47	0.12	310	1.10	1.27	0.76	1.24	1.24	0.73	0.67	1.26	1.26	1.29	0.73	1.28	0.79	1.07	0.99
0.68	0.90	0.20	0.92	0.74	379	1.43	2.30	1.67	1.98	0.00	0.96	1.24	2.13	2.02	2.07	0.00	2.32	1.15	2.22	1.55
0.81	0.41	0.81	0.15	0.17	451	1.52	1.88	1.28	1.41	1.55	0.90	1.15	1.59	1.66	1.42	1.20	1.22	1.05	1.09	0.92
0.70	0.61	0.90	0.89	0.24	467	1.70	2.35	1.49	1.84	0.00	0.00	1.49	2.34	1.89	2.08	1.59	1.83	1.56	1.81	1.29
0.50	0.27	0.73	0.44	0.83	508	1.16	2.10	1.65	1.28	1.79	0.92	1.56	1.55	1.47	1.59	1.56	1.60	1.20	1.43	0.93
0.16	0.21	0.75	0.48	0.98	546	0.00	1.90	1.56	1.00	1.68	0.00	1.60	1.27	0.00	1.47	1.39	1.51	1.17	1.26	0.80
0.91	0.75	0.75	0.24	0.06	563	1.82	2.06	1.32	1.78	1.73	1.01	1.09	0.00	2.01	1.68	1.26	1.51	1.12	1.32	1.21

- The unsupervised objective used to train embeddings and the kind of data it was trained on is important for this dot product method to make sense!
- A neural net version of this (instead of simple dot product):
  - given the learnt embeddings and data about (user, movie) → (rating) that says how much a user liked a movie.
  - we can train with a usual feedforward network using embeddings (either by training back to update the embeddings or not, depending on how much data you have)
  - the input to the network will be the concatenation of the user embedding and the movie embedding
  - maybe we can use a tanh output layer if we would like to express “how much does this user like the movie, etc”. or sigmoid if we want it to be in (0, 1)