Track 2

Caroline Fetzer,
Martin Köppelmann,
Sebastian Stange

Strategy

- Chose relevant attributes by hand (see next slide)
- Generate further attributes using SVD, item-item-similarity, ...
- Create own testsets by sampling over the trainingset
- Use machine learning (Weka) to weight & combine those features
  - Input = trainingset, various own testsets
  - Output = hopefully a good prediction for each user-track-pair in the provided testset

Attributes

user attributes	implemented yet
number of ratings	
number of ratings in each genre	
number of ratings in each genre >= 80	
average rating of the user	
RMSE of users ratings	
track attributes	
number of ratings	x
number of ratings >= 80	x
ratio between number of ratings and number of ratings >= 80	x
number of ratings missing to 20	x
number of genres	x
each genre (true/false)	
RMSE of ratings	"=std?"
std of ratings	x
average rating	x
average preferred genre of users rated this track	
user/track attributes:	
number of tracks rated of the user from the same album	
number of tracks rated of the user from the same genre	
number of tracks rated of the user from the same artist	
number of users rated this song and and another song rated by this user	
distance between the users genre vector and the track genre vector (cosine similarity)	

Results – Error Rates

a) random
   -> **49,9723%**, as expected

b) base prediction (number of ratings for track = Nm)
   -> **42,8856%**

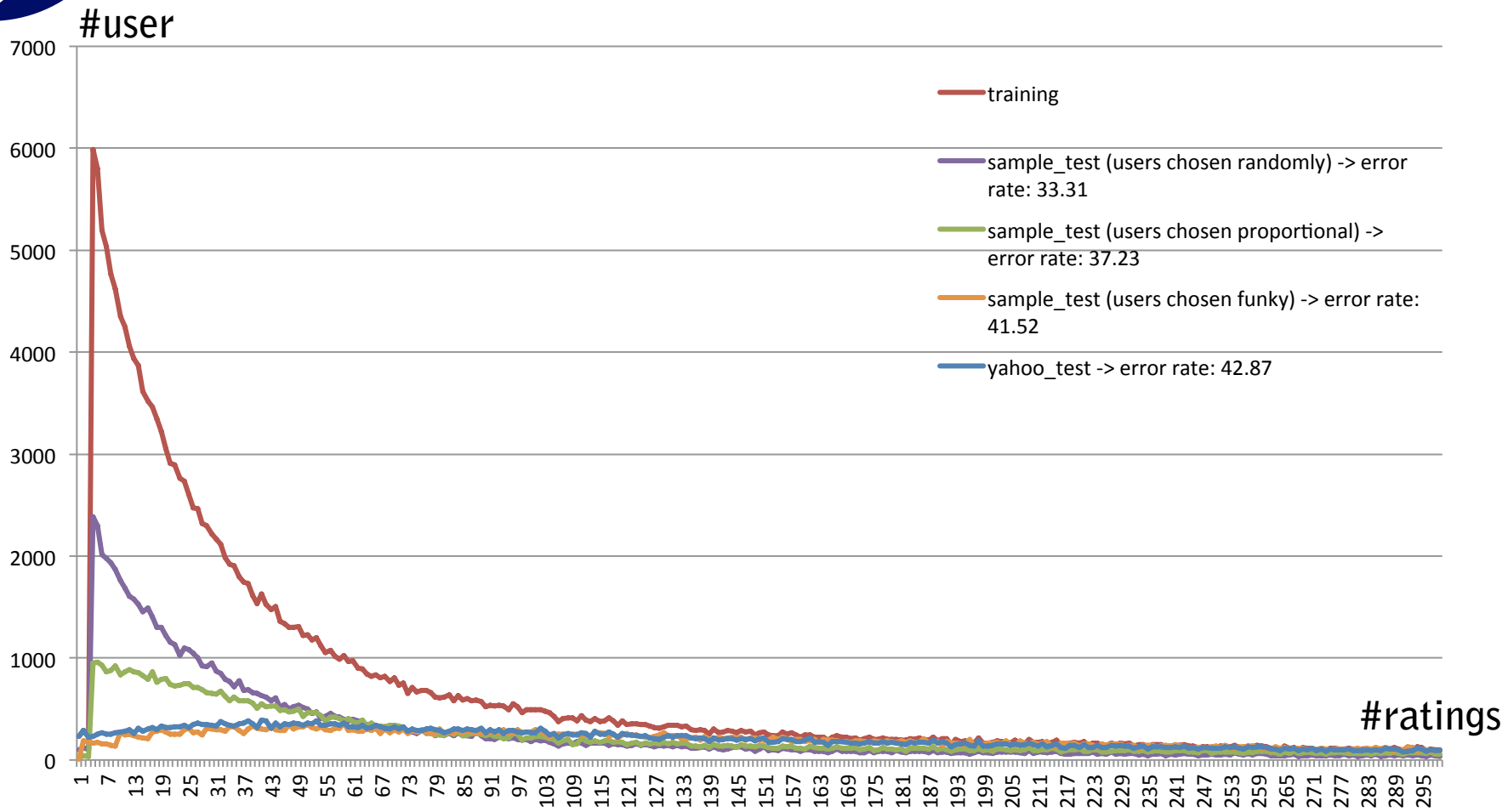
c) restricted base prediction (Nm80 = #(tracks rated >= 80))
   -> **42,8698%**, slightly better than b) indicates that Nm80 is proportional to Nm
   -> yahoo states that unrated tracks in testSet are chosen "proportional to their number of ratings >= 80 in the overall trainingSet"
   → contradiction?
   probably user-sampling effects...

Sampling own testsets

a) chosen unrated tracks randomly
   -> base prediction
   resulted in **4.485%** error rate

b) chosen unrated tracks proportional to high rating occurance
   -> base prediction resulted in **33.3103%** error rate

Sampling own testsets contd.



Questions for other Teams

- Usage of Item-Item similarity?
- Usage of SVD?
- How do you store your data (in memory, DB, ...)?