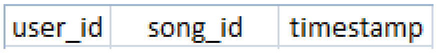
Songs (Tracks) Genre Prediction

1. Provided Problem Statement

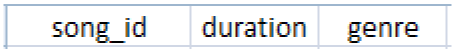
Predict the genre of some tracks with following available data:

session.csv: This file contains records of various user session listening specific tracks of their choice at different point of time. More specifically each record has set of three field user_id, song_id and timestamp.



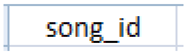
user_id	song_id	timestamp

tracks.csv: It has records of different tracks with their time duration and genre. Each record consists of three field song_id, duration and genre respectively.



song_id	duration	genre

tracks_to_complete.csv: This file contains test data having song_id for which genre needs to be predicted.



song_id

2. Initial Analysis

As per the requirement of problem statement, genre needs to be predicted for the given test data which is only song_id. But initial insights of given data shows some statistics and suggests song_id has no direct definable relation with genre.

Some Statistics:

i) We have *58037* session records in session.csv of 101 users listening *5973* tracks at different timing.

ii) We have *4985* tracks with unique song_ids along with corresponding genre.

iii) We need to predict genre for *1265* tracks provided in tracks_to_complete.csv

iv) There are *1209* extra tracks (song_id) in user session (session.csv) which are not present in tracks.csv thus for them, genre label are not available.

v) There are exactly 1209 tracks common in test set and user session (session.csv)

vi) One of important stat is user sessions contains all the 4985 song_ids which are present in tracks.csv with genre and also contains 1209 song_ids out of 1265 to be predicted.

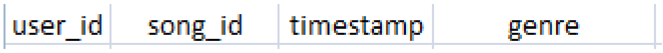
vii) 52 user sessions are not associated with any user_id, can be ignored.

viii) Genre distribution across the song_ids are:

rock	blues	reggae	rap	electro
2406 (48.26%)	807 (16.18%)	785 (15.74%)	593 (11.89%)	394 (7.90%)

Some of the important discovered facts are:

1. Combining tracks data records with session data records basis on song_id



user_id	song_id	timestamp	genre

suggest some correlation *between genres of tracks listen by different users at different point of time* due to:
   - Every user has set of genre choice to listen music tracks by natural interest as human being.
   - User listen different genre related songs at different time schedule i.e. weekend nights may be rock or rap genre are the top choices but daily basis at day time blue genre can be preferred genre choice.
2. song_id in test set (tracks_to_complete.csv) are matched in user session records (session.csv) opens up opportunity to correlate song_id to genre for test data.
3. song_id in test set has no matching with tracks data record (traks.csv) so needs to learn about genre through user's listening behavior.
4. session.csv is most comprehensive data set records which can be used with tracks.csv for ML model preparation (training/verification) and can be used as mediator to predict genre of test set (tracks_to_complete.csv).

Initial analysis motivates to model the relationship between user's listening behavior's characteristics (which user at which time listen which genre type of songs) then figuring out the mechanism to predict genre of a song_id listen by multiple users.

3. Problem Statement Redefined & Solution Pipeline to Work-On

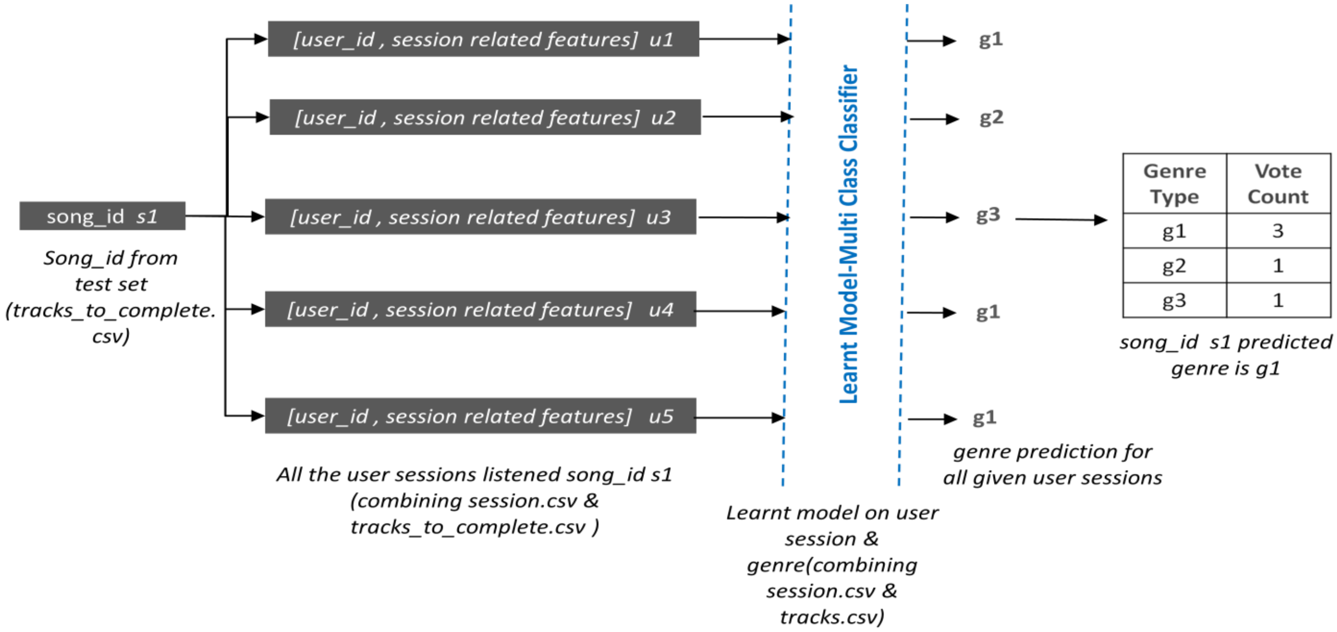
Need to prepare a learning model, multi-class classifier which predict genre for given user with track listening time. Use this model iteratively to determine the predicted genre of song_id. Complete pipelining of solution comprises following steps:

i) Provide the song_id from test data.

ii) Find out all user session records (user_id, timestamp, all other engineered features) listening that song_id.

iii) For each user session record from step ii, get prediction of genre from learnt model.

iv) Declare the genre corresponding to song_id which gets maximum votes.


song_id s1
Song_id from test set (tracks_to_complete. csv)
[user_id , session related features] u1
[user_id , session related features] u2
[user_id , session related features] u3
[user_id , session related features] u4
[user_id , session related features] u5
All the user sessions listened song_id s1 (combining session.csv & tracks_to_complete.csv)
Learnt Model-Multi Class Classifier
Learnt model on user session & genre(combining session.csv & tracks.csv)
g1
g2
g3
g1
g1
genre prediction for all given user sessions
song_id s1 predicted genre is g1

Genre Type	Vote Count
g1	3
g2	1
g3	1

Important Note: There are 1265 song_id listed in test set (tracks_to_complete.csv). Out of that, 1209 song_id which represents tracks can be mapped with user session data (session.csv) and remaining 56 we don't have clue. It means genre prediction for 1209 song_id will work but for remaining 56 tracks, we required to have some heuristics.

4. Solution Pipeline – Machine Learning Method Selection

From the initial analysis, it has been figured out that problem of genre prediction for given dataset is a multi-class classification problem. In order to make further progress towards ML method selection some of the important points to consider are:

i) Set of features are very limited (within the range of 10) e.g. user_id, song_id and timestamp.

ii) Though multiple features can be generated from timestamp but all those created features would highly correlate, not diverse in feature space.

iii) None of the feature qualifies to any value scale rather all seems label features. e.g. user_id, song_id, listening hour or day from timestamp is unique labels.

iv) No. of training examples for derived approach are 46567 by combining session.csv and tracks.csv. It means, we have 46567 unique records comprising user_id, song_id, timestamp (it’s related other features) with label genre.

By considering the all above points and following literature suggestions, it seems we have very limited number of features (even we do perform feature engineering to construct more features), all the features are not so diverse in nature, none of the feature has value scale to vary and limited training examples. So in this case, applying deep learning and its derivates will not suitable choice as machine learning algorithm.

Considering all the constraints of derived data set for learning, we rather chose classical machine learning algorithms which can perform multi-class classification.

5. Solution Pipeline – Feature Engineering

Feature Selection:

As of now, training set consists of just user_id, song_id, duration and timestamp as available features. Duration of the track which seems to have no relation or any justifiable pattern in present context, thus decided to eliminate from feature set.

Feature Creation:

user_id and ***song_id*** are unique representation of record but doesn't have any quantifying measures. Those features serves as categorical in nature and not usable to create new features in provided context.

timestamp shows listening time details of specific tracks by specific user. It has variability in time scale which has good chance to differentiate one user's listening behavior to another with respect to time. By defining different minor level scaling on timestamp value many important features can be created from timestamp.

For the sake of given problem, below are newly created features from timestamp:

year – listening year of a track for a user specified in each training example.

month – listening month (1-12) of a track for a user specified in each training example.

day – listening day (1-30) of a track for a user specified in each training example.

hour – listening hour (0 -23) of a track for a user specified in each training example.

weekday – listening weekday of week (Sunday, Monday, etc.) of a track for a user specified in each training example.

One more feature which can represent mood of the user with respect to different time segments of the day is also created by defining some bins on hour feature.

daypart – listening daypart [LateNight, Morning, Noon, Evening & Night] of a track for a user specified in each training example

Feature Transformation:

user_id and ***song_id*** are transformed as categorical features. time related all the features except ***daypart*** are transformed to numerical and *daypart* feature is transformed as categorical label.

genre label also transformed to numerical category codes as all the classifiers expect labels in numerical or one-hot encoded format.

6. Solution Pipeline – ML Classifier Selection & Model Training-Validation

Reference to step 4 conclusions, Classical machine learning classifier needs to be evaluated based on present paradigm, theoretical aspects and support of some experimentation.

Initially some of the classifier algorithm i.e. Support Vector Machine (SVM), Logistic Regression & Decision Tree based algorithm Random Forest are considered as primary candidature.

Further based on feature's count & their characteristics and experimentation of training/testing with above defined classifier algorithm, it has been observed that Random Forest out performed.

Below is the some rough experimentation metric for all three classifiers:

Classifier	Test Accuracy
SVM	0.60
Logistic Regression	0.35
Random Forest	0.95+

**These experiments are roughly conducted on 70:30 train-validation ratio.*

7. Solution Pipeline – Verification of Performance of Random Forest Classifier Model

There can be chance that trained Random Forest Classifier Model might be

- Over fitted towards training set.
- Validation set remain acute in nature.

In order to test the above mentioned set of hypothesis, further KFold cross validation is performed with number of fold set to 5.

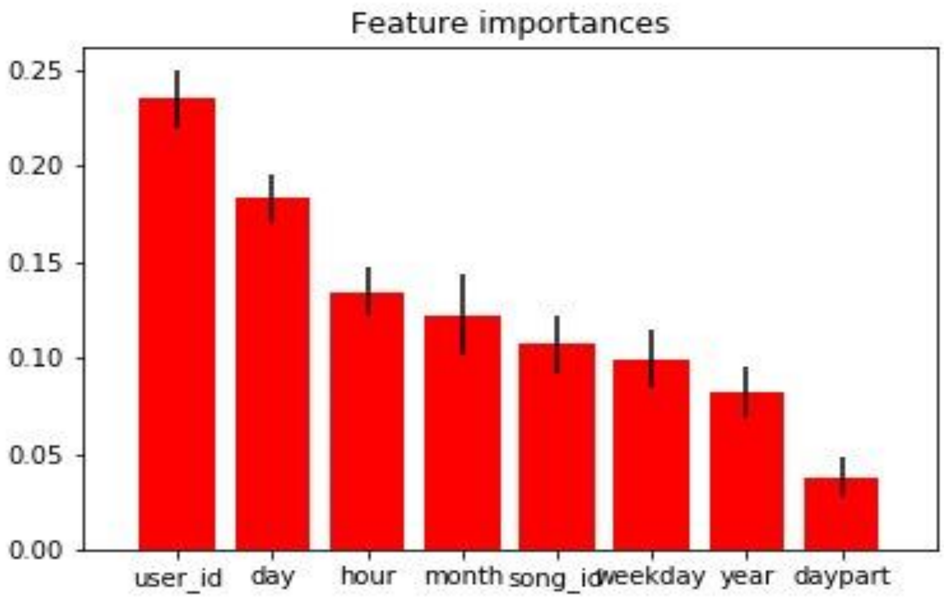
Below are the observed cross validation score:

```
[ 0.96038647  0.96221149  0.95877174  0.9518952   0.95886144]
```

As per the cross validation score, it has been confirmed that classifier performance is consistent with varying set of train – validation data. It also ensure that model is not only out performing for single instance rather performing well *without being over fitted to any specific training set*.

8. Solution Pipeline – Feature Importance

Here is the feature importance calculated of trained *Random Forest Classifier* Model:



Feature Ranking:

Feature Name	**Rank (Dominance)**
user_id	1 (0.23)
day	2 (0.18)
hour	3 (0.13)
month	4 (0.12)
song_id	5 (0.10)
weekday	6 (0.09)
year	7 (0.08)
daypart	8 (0.03)

Feature ranking and their dominance distribution shows that newly created features also contributed in much better way to have robust classifier performance even though user_id remain prominent one. Lastly created feature does not show any significant contribution.

9. Solution Pipeline – Genre Prediction of Provided Validation Set (tracks_to_complete.csv)

With reference to step 7 and 8, it has been ensured that Random Forest Classifier performing well (95% – 96% accuracy) without being over fit as we can believe on 5 fold cross validation. So at this step, Random Forest Classifier gets trained on complete training set (46567 training example) and predict the genre of given test set.

- On test set prediction are made of all the test set examples then predicted label is assigned to the genre which occurs more time than other as mentioned in step 3.
- All the 56 tracks for which user session record were not available are assigned to rock genre because rock is most occurring genre (from step 2).

Predicted Genre distribution for Test Set (traks_to_complete.csv)

rock	blues	reggae	rap	Electro
614 (48.53%)	200 (15.81%)	207 (16.36%)	145 (11.6%)	99 (7.82%)

Predicted Genre Result's Justification

As it is mentioned and weakly assumed that present problem's data is generated and then may be divided into training purpose (tracks.csv and session.csv) and validation purpose (tracks_to_ complete.csv). In other words, train data & validation data derived from same distribution then their respective genre distribution also follow the similar pattern.

By referencing the train data's genre distribution from step 2 table and comparing it with valida tion data's genre distribution clearly shows that both are very closely matches to each other (re fer below graph). This fact justifies the correctness of predicted genre up to a remarkable level.

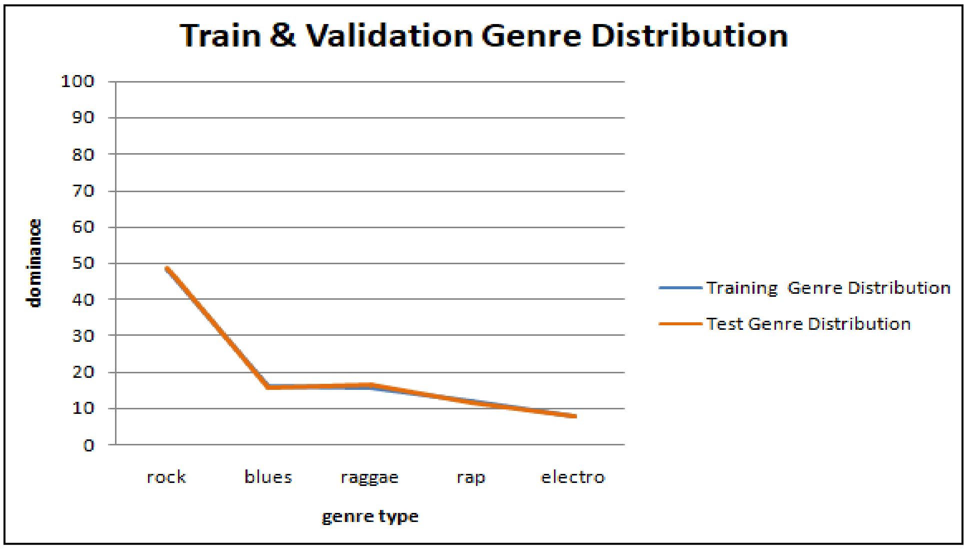


Figure 1: Test Genre completely following Train genre distribution

Predicted genre along with song_id are saved in solution.csv.

10. Solution Pipeline – Further Improvement Scope

As predication of genre has been concluded and results are achieved with justification of correctness. There are some points and direction which can be referred towards improvement of overall solution a s follows:

i) Parameter tuning of selected ML classifier algorithm
ii) Features normalization for numerical features e.g. one hot encoding
iii) New feature creation & removal of unnecessary features.
iv) etc.

11. Solution Pipeline – Technologies/Libraries/Language

Programming Language: python
Libraries: numpy, pandas, matplotlib
ML Library: scikit-learn