Name That Genre

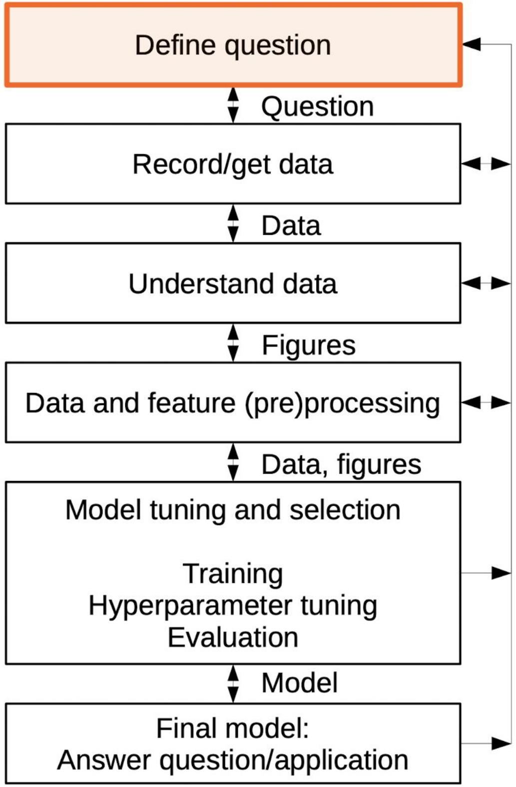
Spotify Genre Classification Using Machine Learning

Ciesla, Hörschinger, Oberascher

Define question and goal

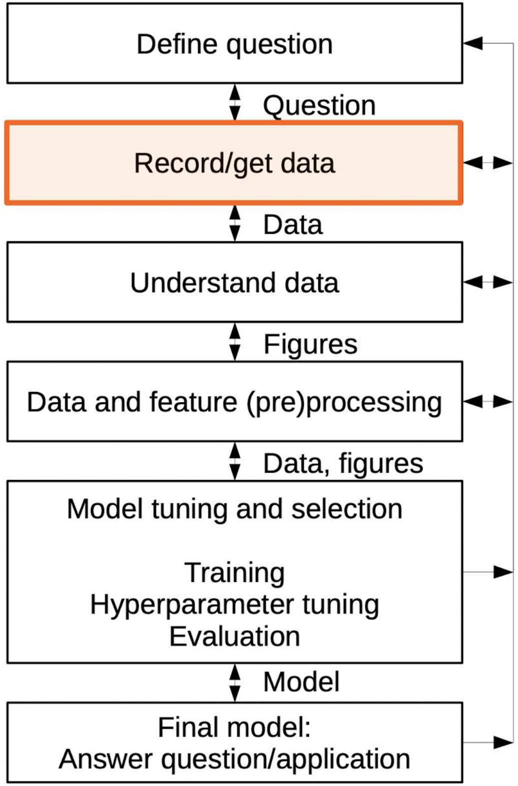
- Predict Genre of Songs
- Evaluate Performances of different ML models

Spotify for Developers



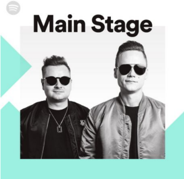
Record Data - Overview

- $ pip install ***spotipy***
- create spotify developer account
- **playlist_items**(playlist_id) -> get all track IDs from playlist
- **audio_features**([track_ids]) -> get list of audio features from list of track IDs
- save features to .json file

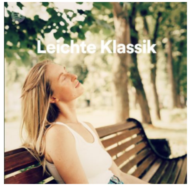


Record Data - Labeling

jazz



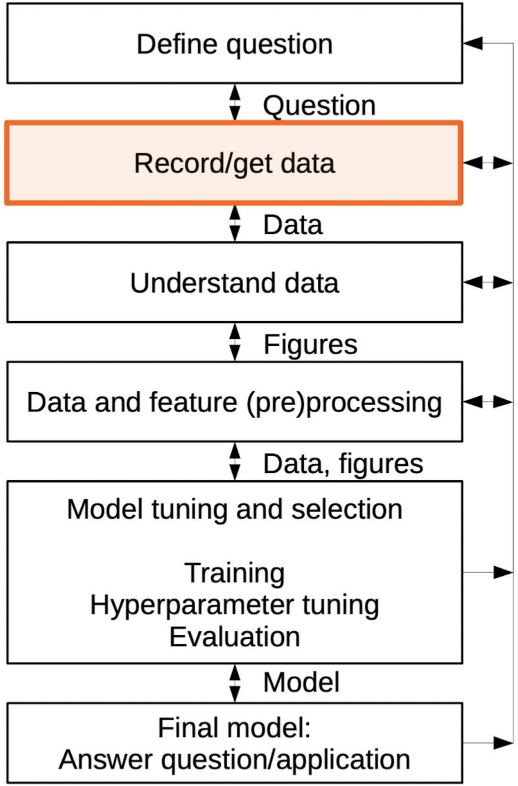
edm



classic

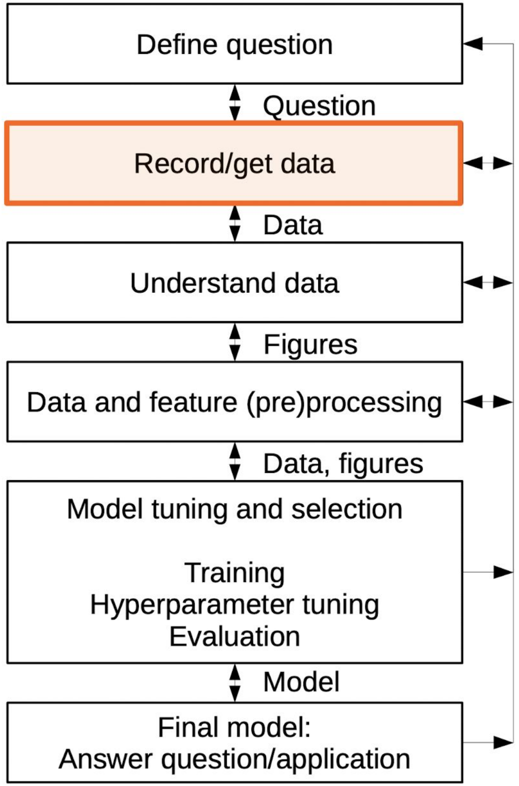
rock

hiphop



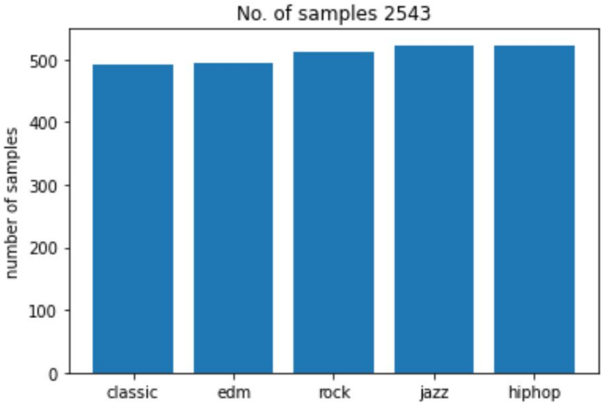
Record Data - Audio Features

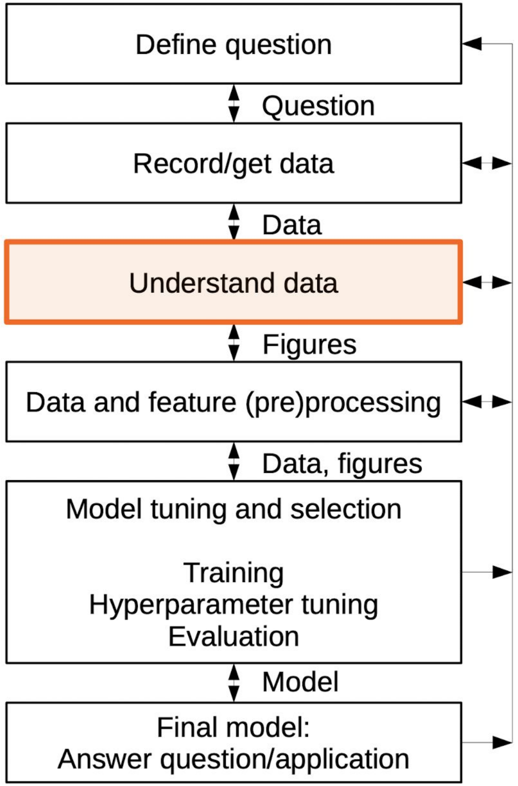

```
{ ⊟
    "danceability":0.194,
    "energy":0.0324,
    "key":5,
    "loudness":-28.215,
    "mode":1,
    "speechiness":0.0382,
    "acousticness":0.982,
    "instrumentalness":0.961,
    "liveness":0.0916,
    "valence":0.0596,
    "tempo":144.13,
    "type":"audio_features",
    "id":"2YarjDYjBJuH63dUIh9OWv",
    "uri":"spotify:track:2YarjDYjBJuH63dUIh9OWv",
    "track_href":"https://api.spotify.com/v1/tracks/2YarjDYjBJuH63dUIh9OWv",
    "analysis_url":"https://api.spotify.com/v1/audio-analysis/2YarjDYjBJuH63dUIh9OWv",
    "duration_ms":433800,
    "time_signature":4,
    "genre":"classic",
    "playlist_id":"37i9dQZF1DXaHEllsiT8lf"
},
```

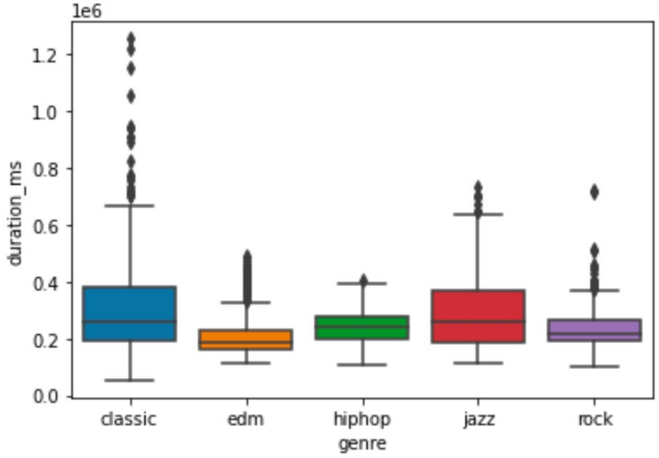
Understand Data

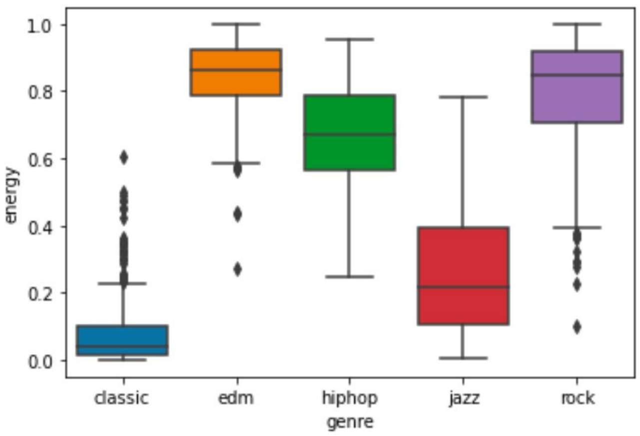
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2580 entries, 0 to 2579
Data columns (total 15 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   danceability      2580 non-null   float64
 1   energy            2580 non-null   float64
 2   key               2580 non-null   int64
 3   loudness          2580 non-null   float64
 4   mode              2580 non-null   int64
 5   speechiness       2580 non-null   float64
 6   acousticness      2580 non-null   float64
 7   instrumentalness  2580 non-null   float64
 8   liveness          2580 non-null   float64
 9   valence           2580 non-null   float64
 10  tempo             2580 non-null   float64
 11  duration_ms       2580 non-null   int64
 12  time_signature    2580 non-null   int64
 13  genre             2580 non-null   object
 14  playlist_id       2580 non-null   object
dtypes: float64(9), int64(4), object(2)
memory usage: 302.5+ KB
```

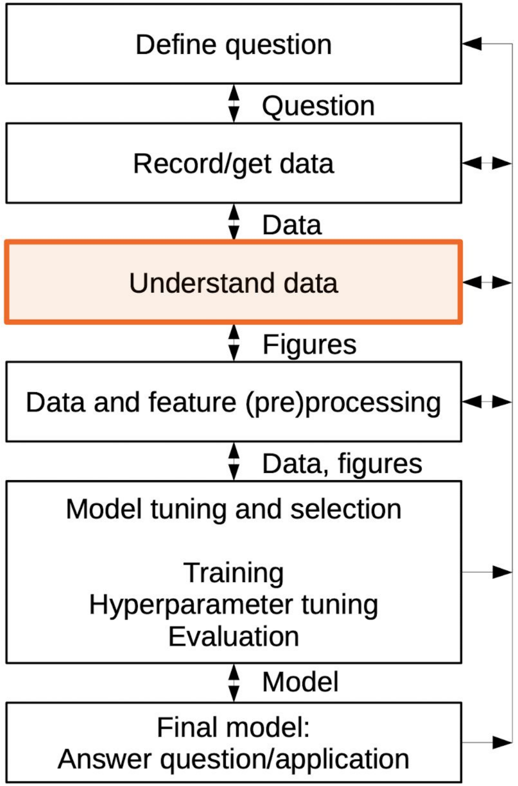


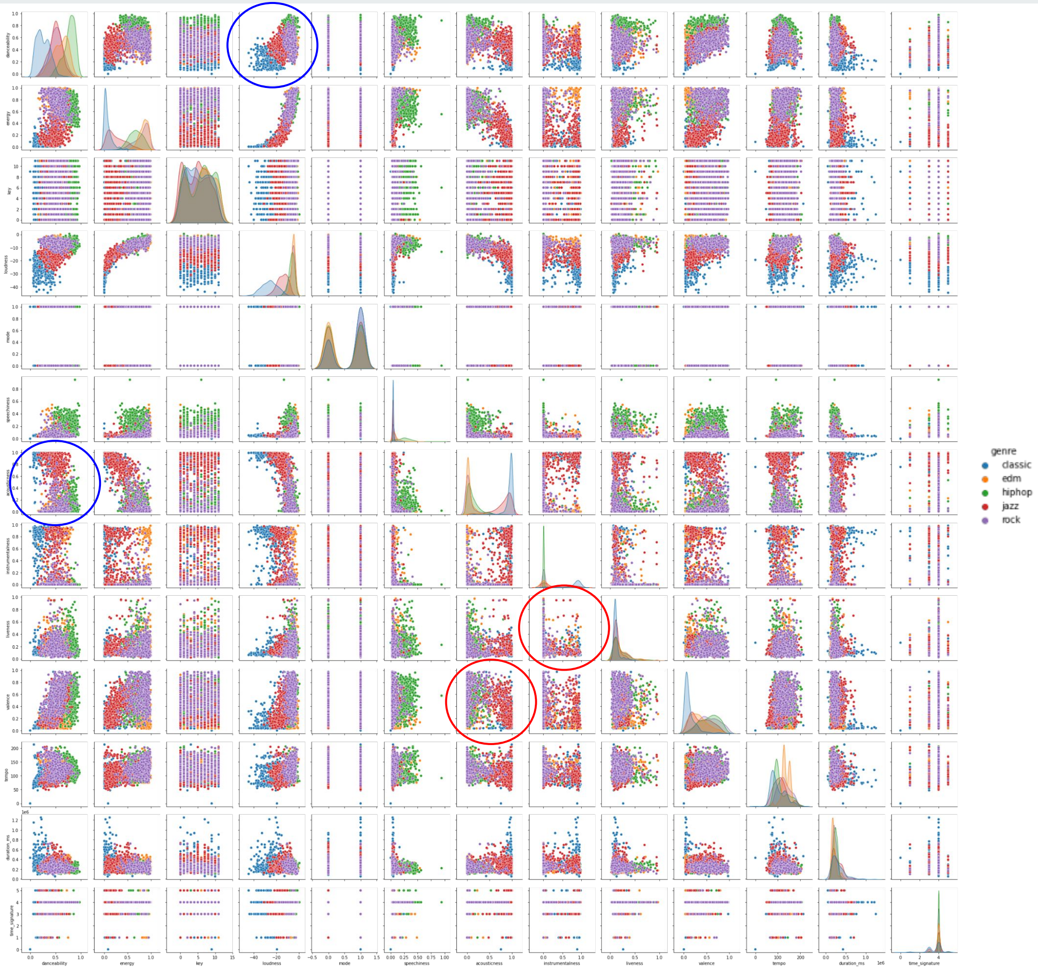


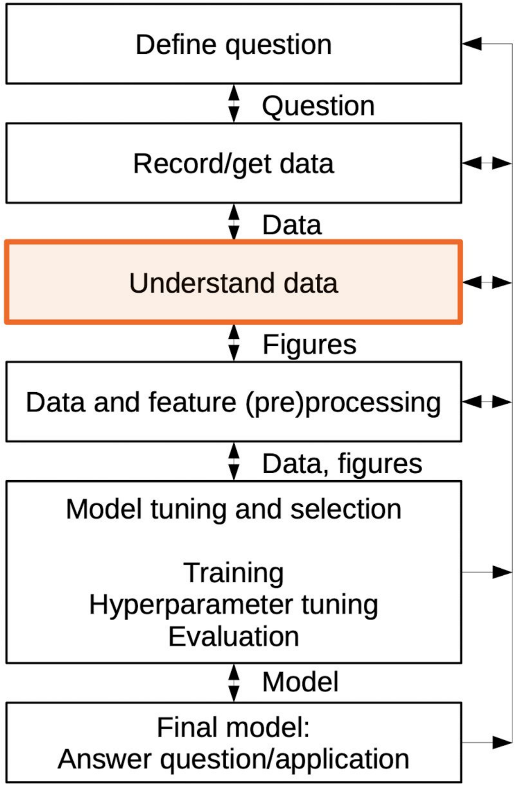
Understand Data



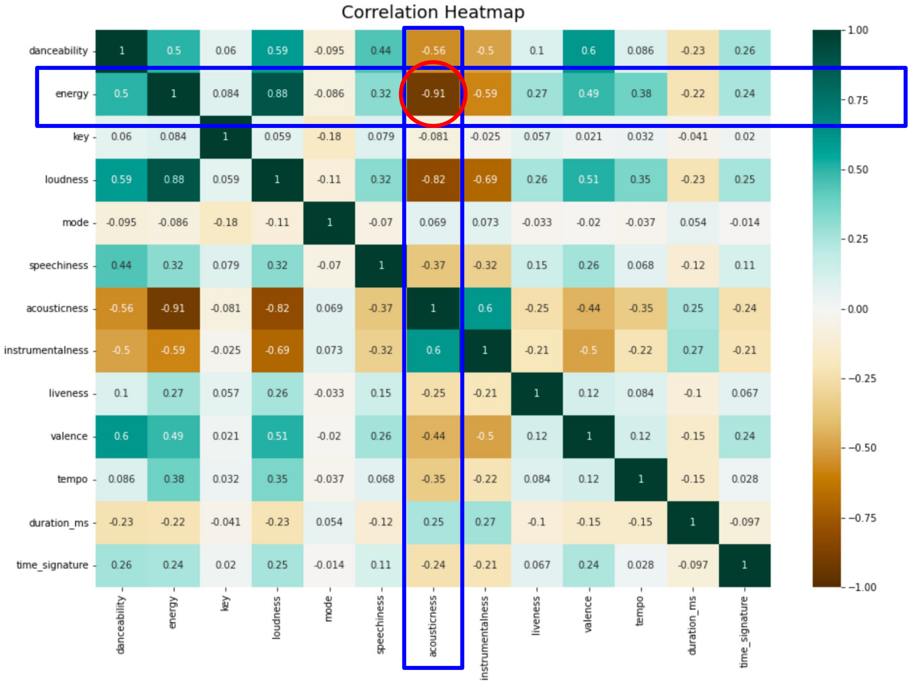


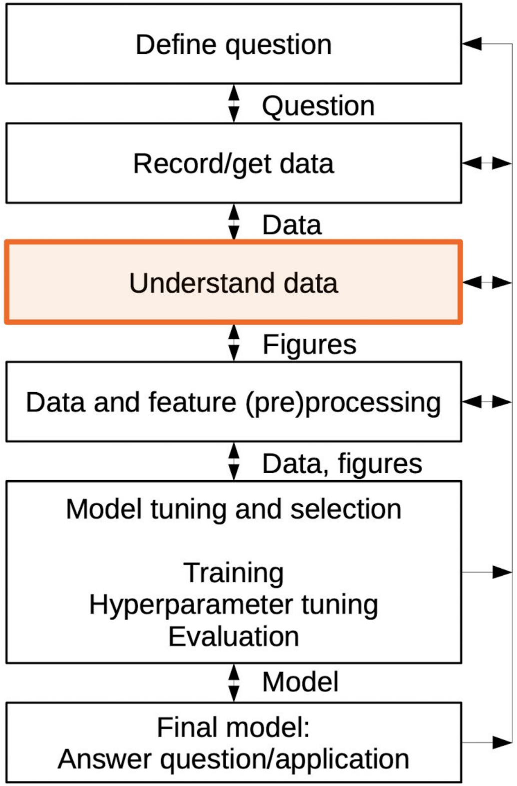



genre
classic
edm
hiphop
jazz
rock
danceability
energy
key
loudness
mode
speechiness
acousticness
instrumentalness
liveness
valence
tempo
duration_ms
time_signature


Define question
Question
Record/get data
Data
Understand data
Figures
Data and feature (pre)processing
Data, figures
Model tuning and selection
Training
Hyperparameter tuning
Evaluation
Model
Final model:
Answer question/application

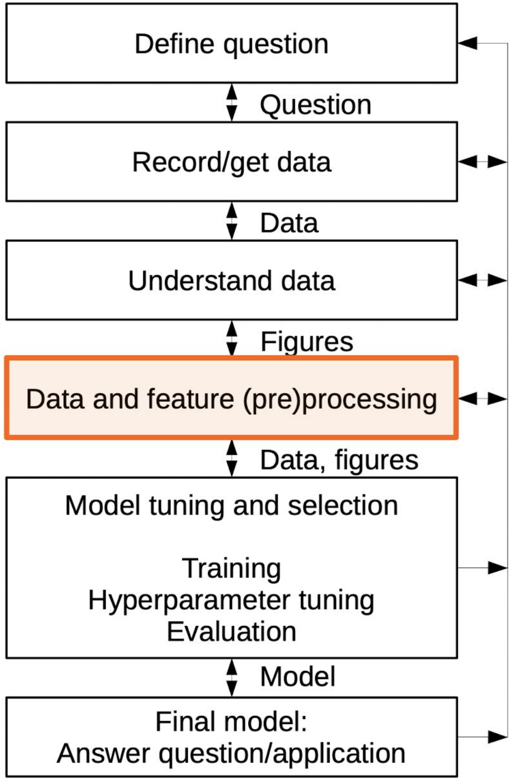
Understand Data - Correlation





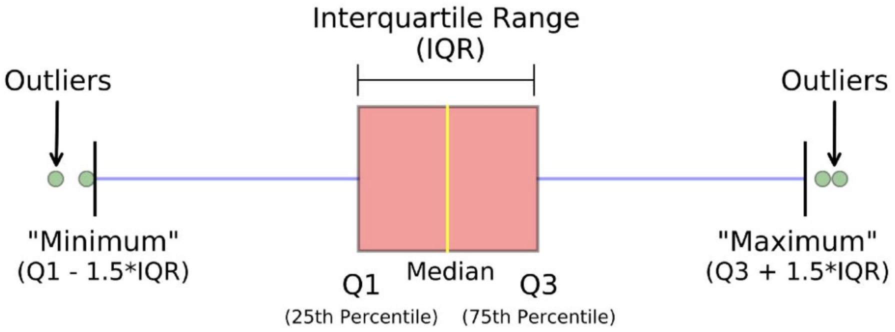
Data and Feature Preprocessing

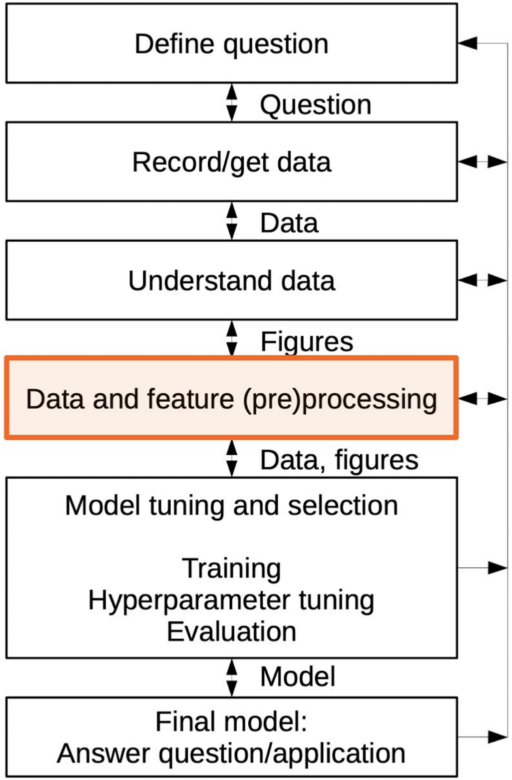
- Remove invalid samples
  - some samples with wrong key and wrong time signature
  - key == -1 or time_signature not in [3,7]
- Reduce highly correlated features
  - energy & loudness
- Scale numerical features to mean = 0 & standard deviation = 1



Data and Feature Preprocessing

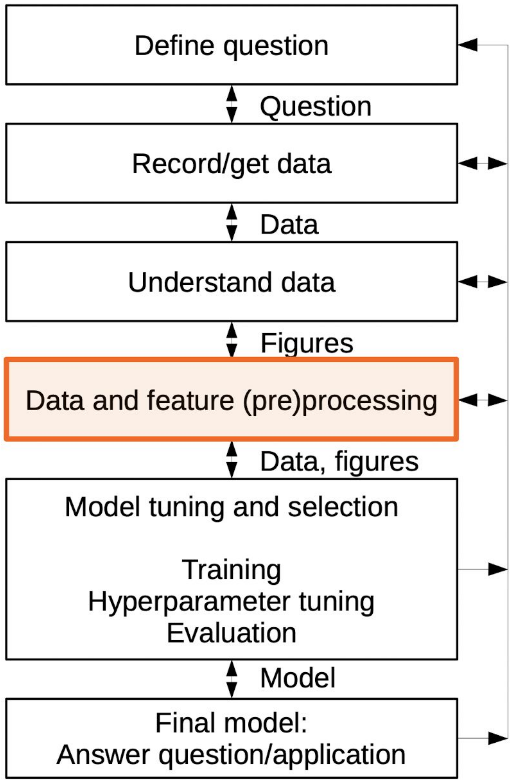
- Restrict outliers
  - upper whisker → Q3 + 1.5*IQR
  - lower whisker → Q1 - 1.5*IQR



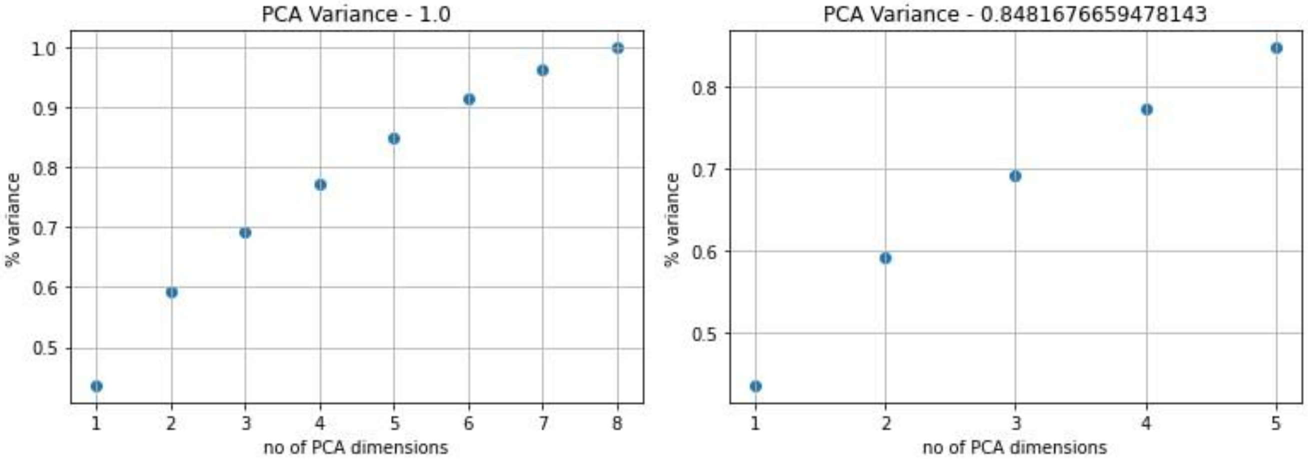


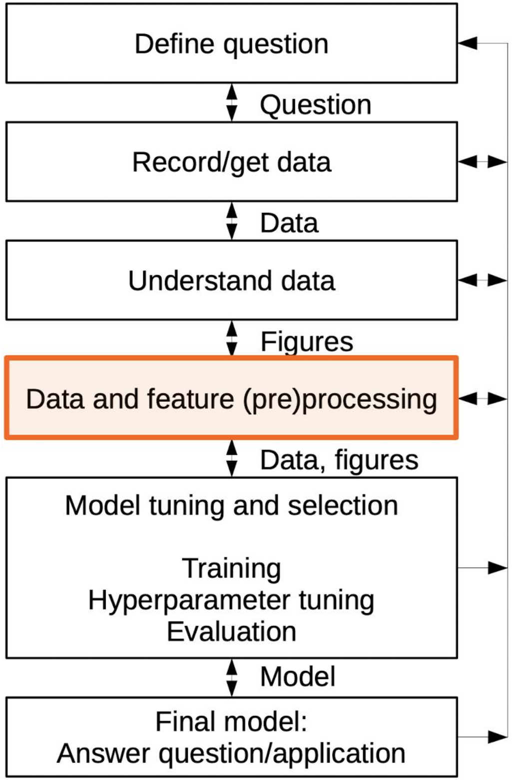
Data and Feature Preprocessing

- One-hot-encoding for categorical features
- **Final features**
  - numerical → danceability, speechiness, acousticness, instrumentalness, liveness, valence, tempo, duration_ms
  - categorical → mode (binary), time_signature_0 - time-signature_5, key_0 - key_11
- Perform preprocessing steps on training set
- Perform **same** preprocessing steps on test set



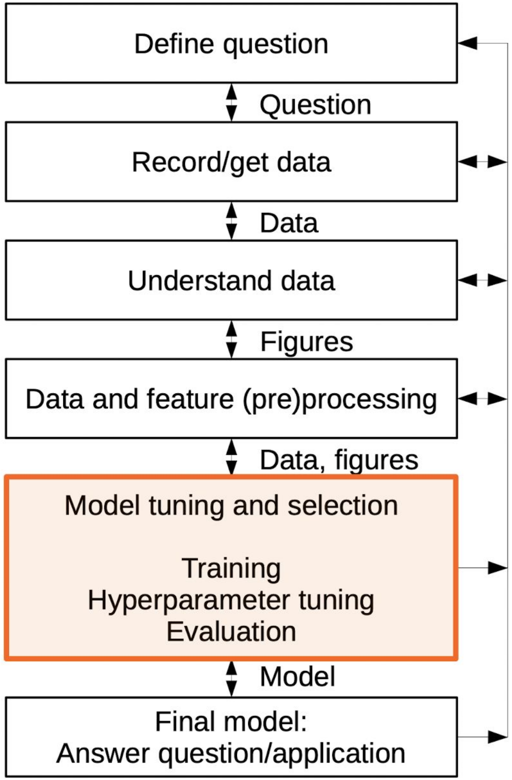
Data and Feature Preprocessing - PCA





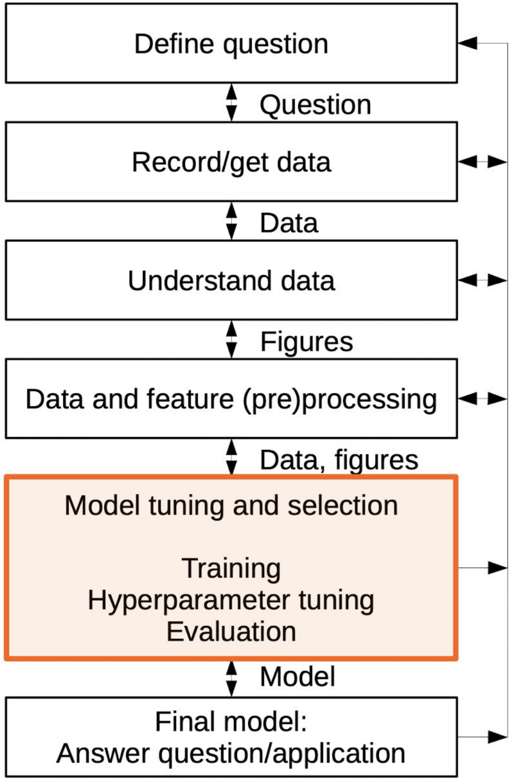
Model tuning

- Nested Cross Validation
  - different data for hyperparameter tuning and model performance evaluation
  - for every estimator (KNN, SVC & Random Forest)
    - inner loop
      - 5-fold CV
      - grid search for hyperparameter tuning
    - outer loop → model performance evaluation
      - 5-fold CV



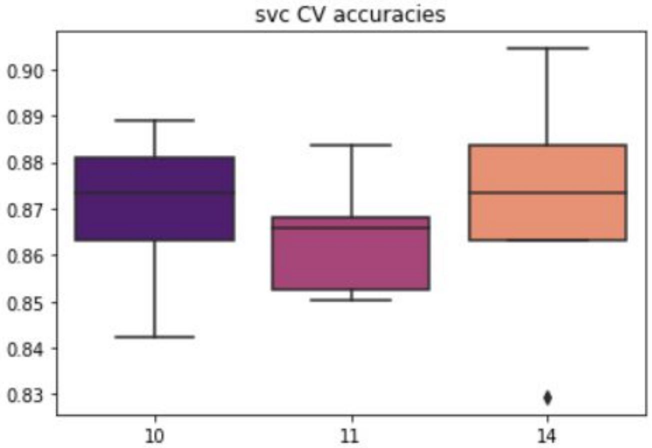
Model selection

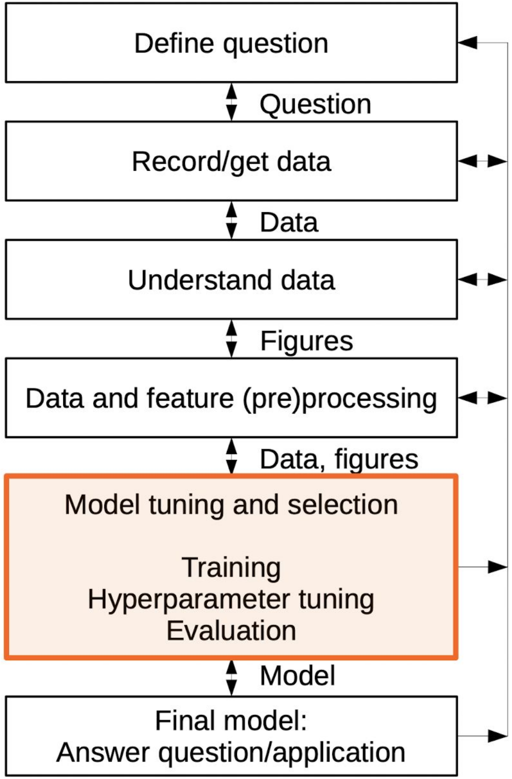
	model	params	nested_cv_training_acc	cv_training_acc	test_acc	test_roc_auc
0	knn	{"metric": "manhattan", "n_neighbors": 23, "we...	0.878553	0.857364	0.852713	0.980116
1	knn	{"metric": "euclidean", "n_neighbors": 26, "we...	0.842377	0.852196	0.855814	0.979187
2	knn	{"metric": "manhattan", "n_neighbors": 33, "we...	0.860465	0.859432	0.855814	0.981063
3	knn	{"metric": "manhattan", "n_neighbors": 16, "we...	0.832041	0.862016	0.854264	0.979672
4	knn	{"metric": "manhattan", "n_neighbors": 16, "we...	0.873385	0.862016	0.854264	0.979672
5	randomForest	{"max_depth": 40, "max_features": 5, "min_samp...	0.891473	0.893023	0.888372	0.987955
6	randomForest	{"max_depth": 30, "max_features": 5, "min_samp...	0.870801	0.896124	0.889922	0.988184
7	randomForest	{"max_depth": 30, "max_features": 5, "min_samp...	0.909561	0.888889	0.883721	0.987549
8	randomForest	{"max_depth": 60, "max_features": 5, "min_samp...	0.883721	0.895090	0.880620	0.987339
9	randomForest	{"max_depth": 30, "max_features": 10, "min_sam...	0.912145	0.893540	0.882171	0.987563
10	svc	{"C": 3.0, "gamma": 0.1, "kernel": "rbf"}	0.873385	0.869767	0.891473	0.983729
11	svc	{"C": 1.0, "gamma": 0.1, "kernel": "rbf"}	0.844961	0.864083	0.886822	0.982449
12	svc	{"C": 3.0, "gamma": 0.1, "kernel": "rbf"}	0.886305	0.869767	0.891473	0.983664
13	svc	{"C": 3.0, "gamma": 0.1, "kernel": "rbf"}	0.847545	0.869767	0.891473	0.983746
14	svc	{"C": 9.0, "gamma": 0.1, "kernel": "rbf"}	0.894057	0.870801	0.896124	0.984762



Model selection - SVC

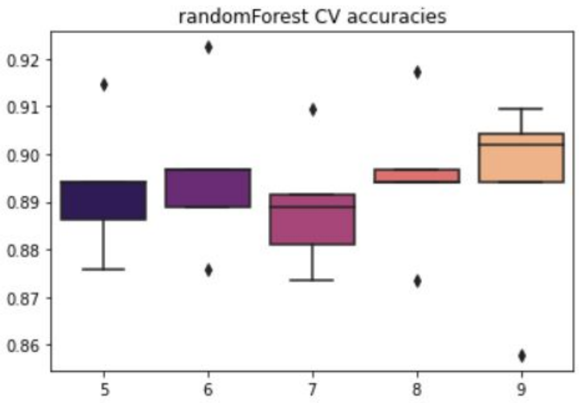
```
10: {"C": 3.0, "gamma": 0.1, "kernel": "rbf"}
11: {"C": 1.0, "gamma": 0.1, "kernel": "rbf"}
14: {"C": 9.0, "gamma": 0.1, "kernel": "rbf"}
```

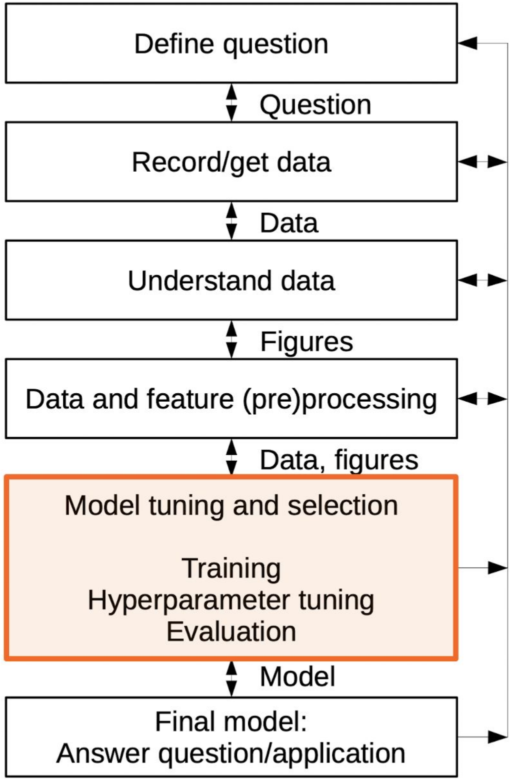




Model selection - RF

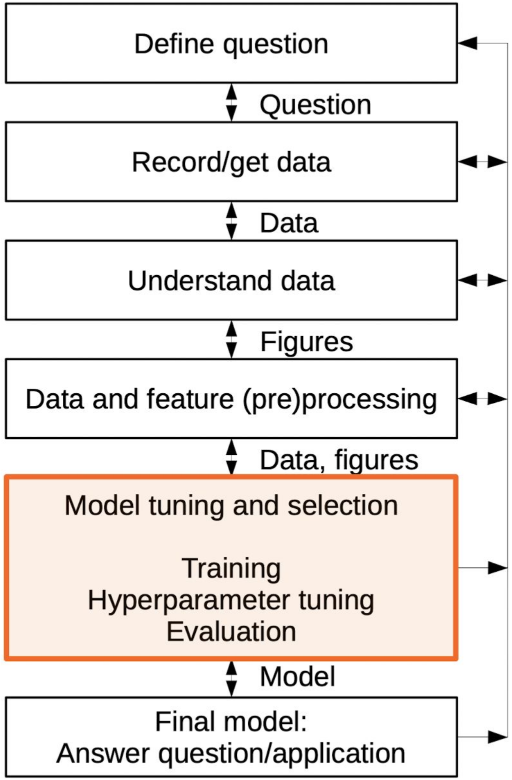
```
5: {"max_depth": 40, "max_features": 5, "min_samples_leaf": 1, "min_samples_split": 5, "n_estimators": 2000}
6: {"max_depth": 30, "max_features": 5, "min_samples_leaf": 1, "min_samples_split": 3, "n_estimators": 4000}
7: {"max_depth": 30, "max_features": 5, "min_samples_leaf": 2, "min_samples_split": 3, "n_estimators": 4000}
8: {"max_depth": 60, "max_features": 5, "min_samples_leaf": 2, "min_samples_split": 3, "n_estimators": 1000}
9: {"max_depth": 30, "max_features": 10, "min_samples_leaf": 1, "min_samples_split": 3, "n_estimators": 4000}
```



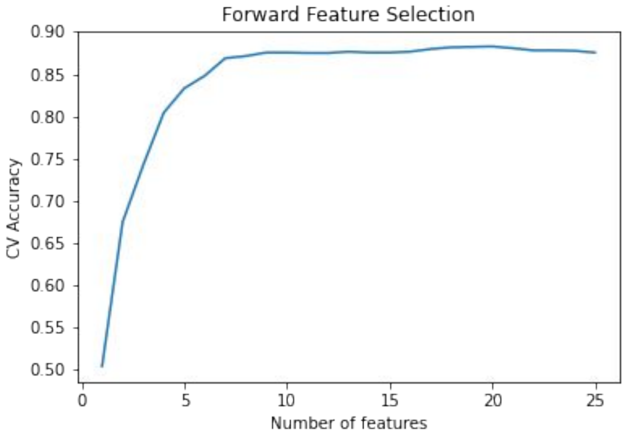


Feature selection - Forward Selection

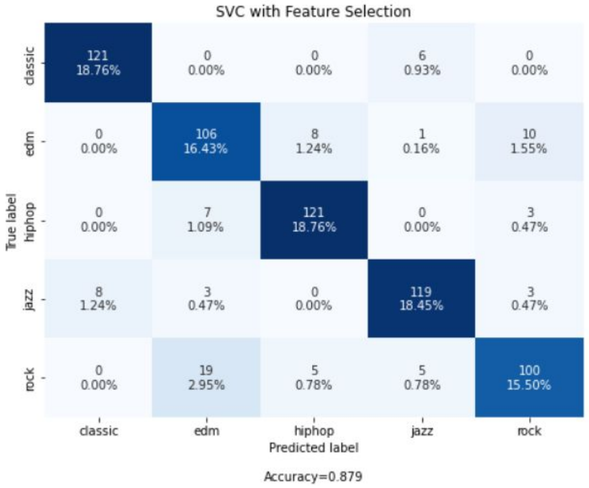
CV Accuracy	Nr.	Feature Names
0.504	1	acousticness
0.675	2	acousticness, danceability
0.742	3	acousticness, danceability, tempo
0.804	4	acousticness, danceability, tempo, valence
0.833	5	acousticness, danceability, tempo, valence, instrumentalness
0.848	6	acousticness, danceability, tempo, valence, instrumentalness, speechiness
0.869	7	acousticness, danceability, tempo, valence, instrumentalness, speechiness, duration_ms
0.871	8	acousticness, danceability, tempo, valence, instrumentalness, speechiness, duration_ms, key_4
0.875	9	acousticness, danceability, tempo, valence, instrumentalness, speechiness, duration_ms, key_4, key_11



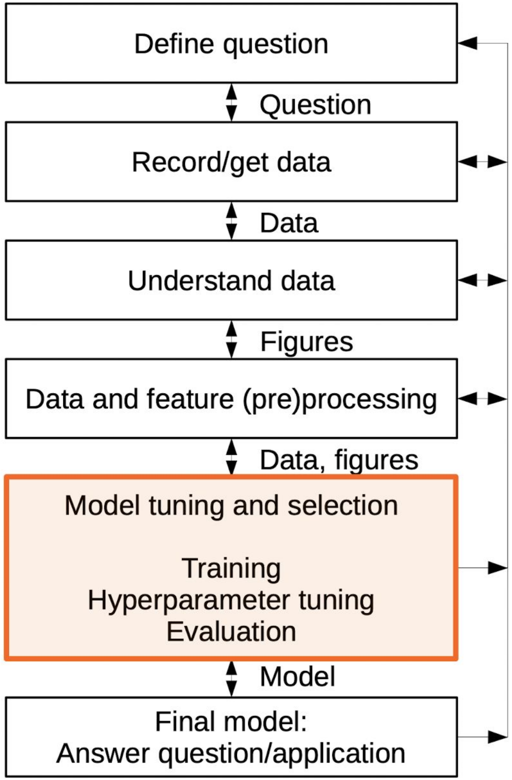
Feature selection - Forward Selection



max = 0.883 (with 20 features)

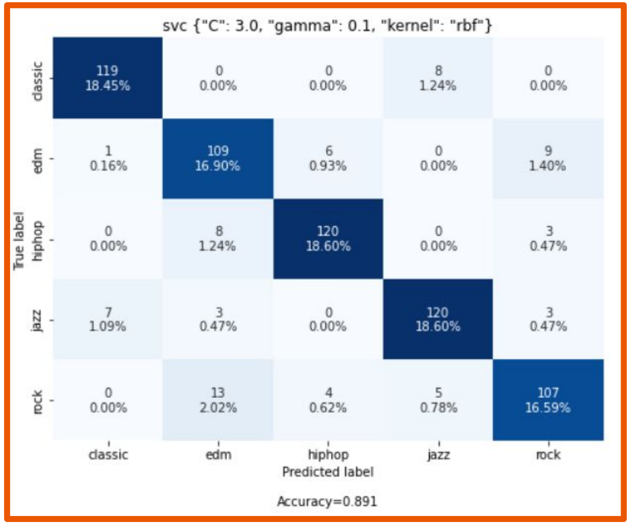


with 9 features

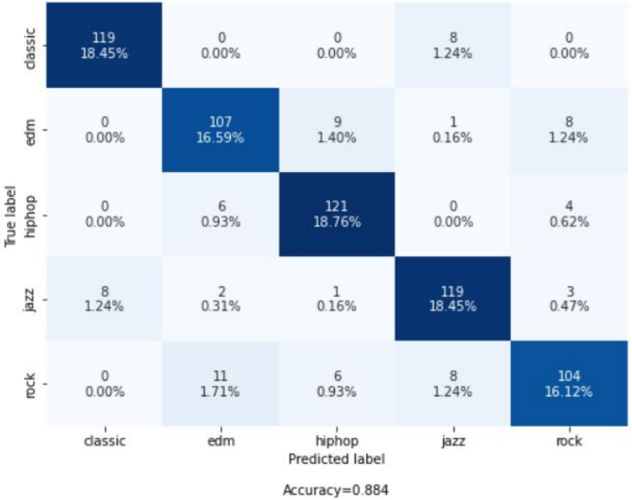


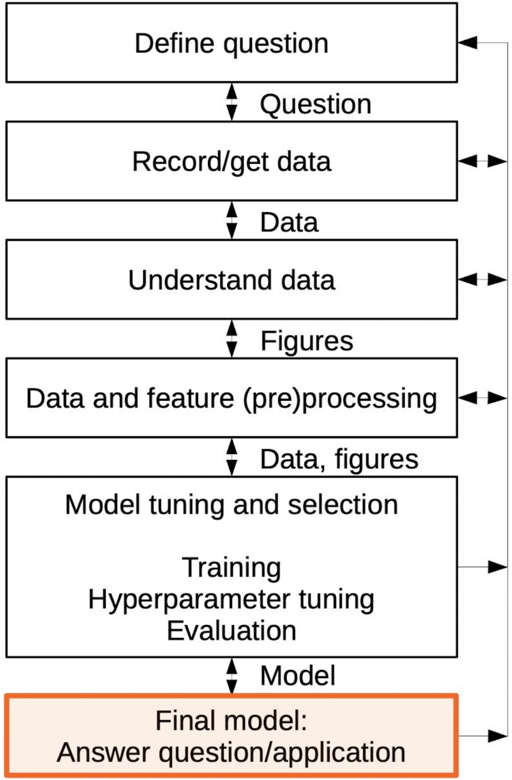
Model selection → SVM

SVC



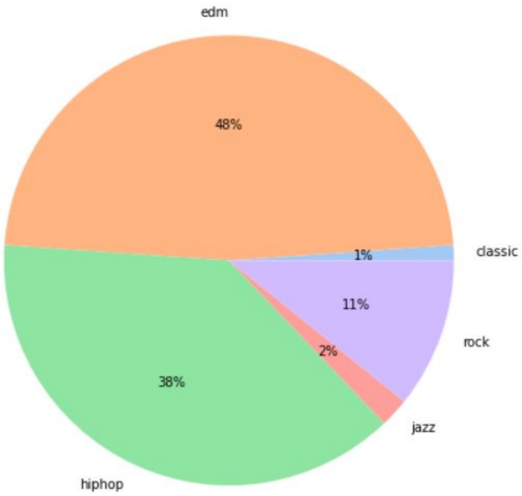
RF



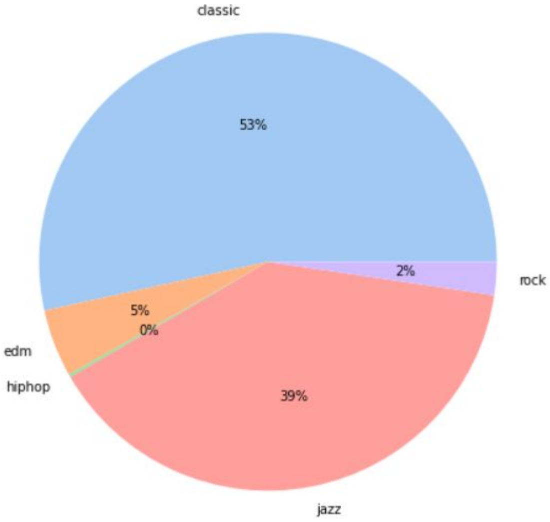


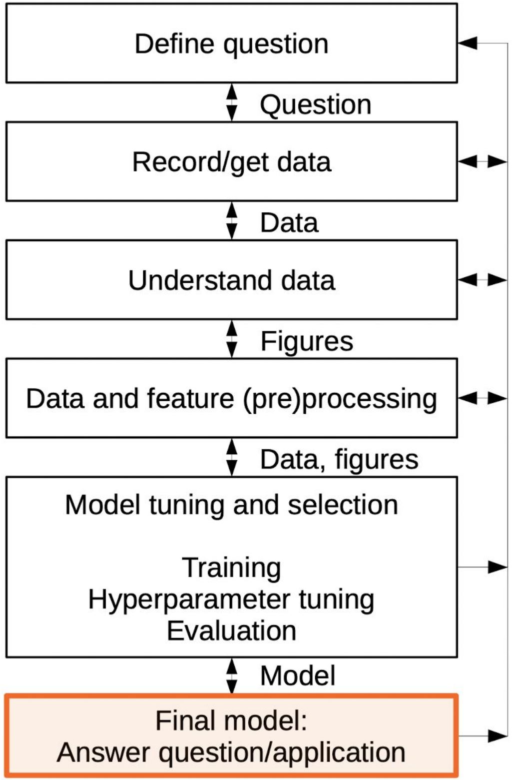
Final model - answer the question

Lose Yourself - Eminem
hiphop → edm



Dear Kathy - Benny Golson
jazz → classic





Thank you for your attention!