Project 3 - Recommender Systems

In [1]:

```
!pip install regex
!pip install nltk
!pip install sklearn
!pip install umap-learn[plot]
!pip install holoviews
!pip install -U ipykernel
!pip install scikit-surprise

import sys
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import regex as re
import random
import nltk
import multiprocessing
import umap.umap_ as umap
import umap.plot
import re
import string
import warnings

from sklearn.metrics import roc_curve, auc, mean_squared_error
from surprise import Reader, Dataset, accuracy
from surprise.prediction_algorithms.knns import KNNWithMeans
from surprise.model_selection import cross_validate, KFold, train_test_split
from surprise.prediction_algorithms.matrix_factorization import NMF, SVD

np.random.seed(0)
random.seed(0)
```

```
Requirement already satisfied: regex in /Users/madhavsankar/opt/anaconda3/lib/py
thon3.9/site-packages (2021.8.3)
Requirement already satisfied: nltk in /Users/madhavsankar/opt/anaconda3/lib/pyt
hon3.9/site-packages (3.6.5)
Requirement already satisfied: click in /Users/madhavsankar/opt/anaconda3/lib/py
thon3.9/site-packages (from nltk) (8.0.3)
Requirement already satisfied: joblib in /Users/madhavsankar/opt/anaconda3/lib/p
ython3.9/site-packages (from nltk) (1.1.0)
Requirement already satisfied: regex>=2021.8.3 in /Users/madhavsankar/opt/anacon
da3/lib/python3.9/site-packages (from nltk) (2021.8.3)
Requirement already satisfied: tqdm in /Users/madhavsankar/opt/anaconda3/lib/pyt
hon3.9/site-packages (from nltk) (4.62.3)
Requirement already satisfied: sklearn in /Users/madhavsankar/opt/anaconda3/lib/
python3.9/site-packages (0.0)
Requirement already satisfied: scikit-learn in /Users/madhavsankar/opt/anaconda
3/lib/python3.9/site-packages (from sklearn) (0.24.2)
Requirement already satisfied: joblib>=0.11 in /Users/madhavsankar/opt/anaconda
3/lib/python3.9/site-packages (from scikit-learn->sklearn) (1.1.0)
Requirement already satisfied: scipy>=0.19.1 in /Users/madhavsankar/opt/anaconda
3/lib/python3.9/site-packages (from scikit-learn->sklearn) (1.7.1)
Requirement already satisfied: numpy>=1.13.3 in /Users/madhavsankar/opt/anaconda
3/lib/python3.9/site-packages (from scikit-learn->sklearn) (1.20.3)
Requirement already satisfied: threadpoolctl>=2.0.0 in /Users/madhavsankar/opt/a
```

```
naconda3/lib/python3.9/site-packages (from scikit-learn->sklearn) (2.2.0)
zsh:1: no matches found: umap-learn[plot]
Requirement already satisfied: holoviews in /Users/madhavsankar/opt/anaconda3/li
b/python3.9/site-packages (1.12.7)
Requirement already satisfied: numpy>=1.0 in /Users/madhavsankar/opt/anaconda3/l
ib/python3.9/site-packages (from holoviews) (1.20.3)
Requirement already satisfied: param<2.0,>=1.8.0 in /Users/madhavsankar/opt/anac
onda3/lib/python3.9/site-packages (from holoviews) (1.12.0)
Requirement already satisfied: pyviz-comms>=0.7.2 in /Users/madhavsankar/opt/ana
conda3/lib/python3.9/site-packages (from holoviews) (2.0.2)
Requirement already satisfied: ipykernel in /Users/madhavsankar/opt/anaconda3/li
b/python3.9/site-packages (6.9.1)
Requirement already satisfied: matplotlib-inline<0.2.0,>=0.1.0 in /Users/madhavs
ankar/opt/anaconda3/lib/python3.9/site-packages (from ipykernel) (0.1.2)
Requirement already satisfied: ipython>=7.23.1 in /Users/madhavsankar/opt/anacon
da3/lib/python3.9/site-packages (from ipykernel) (7.29.0)
Requirement already satisfied: appnope in /Users/madhavsankar/opt/anaconda3/lib/
python3.9/site-packages (from ipykernel) (0.1.2)
Requirement already satisfied: tornado<7.0,>=4.2 in /Users/madhavsankar/opt/anac
onda3/lib/python3.9/site-packages (from ipykernel) (6.1)
Requirement already satisfied: traitlets<6.0,>=5.1.0 in /Users/madhavsankar/opt/
anaconda3/lib/python3.9/site-packages (from ipykernel) (5.1.0)
Requirement already satisfied: debugpy<2.0,>=1.0.0 in /Users/madhavsankar/opt/an
aconda3/lib/python3.9/site-packages (from ipykernel) (1.4.1)
Requirement already satisfied: jupyter-client<8.0 in /Users/madhavsankar/opt/ana
conda3/lib/python3.9/site-packages (from ipykernel) (6.1.12)
Requirement already satisfied: nest-asyncio in /Users/madhavsankar/opt/anaconda
3/lib/python3.9/site-packages (from ipykernel) (1.5.1)
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0 in /
Users/madhavsankar/opt/anaconda3/lib/python3.9/site-packages (from ipython>=7.2
3.1->ipykernel) (3.0.20)
Requirement already satisfied: backcall in /Users/madhavsankar/opt/anaconda3/li
b/python3.9/site-packages (from ipython>=7.23.1->ipykernel) (0.2.0)
Requirement already satisfied: pygments in /Users/madhavsankar/opt/anaconda3/li
b/python3.9/site-packages (from ipython>=7.23.1->ipykernel) (2.10.0)
Requirement already satisfied: setuptools>=18.5 in /Users/madhavsankar/opt/anaco
nda3/lib/python3.9/site-packages (from ipython>=7.23.1->ipykernel) (58.0.4)
Requirement already satisfied: jedi>=0.16 in /Users/madhavsankar/opt/anaconda3/l
ib/python3.9/site-packages (from ipython>=7.23.1->ipykernel) (0.18.0)
Requirement already satisfied: decorator in /Users/madhavsankar/opt/anaconda3/li
b/python3.9/site-packages (from ipython>=7.23.1->ipykernel) (5.1.0)
Requirement already satisfied: pickleshare in /Users/madhavsankar/opt/anaconda3/
lib/python3.9/site-packages (from ipython>=7.23.1->ipykernel) (0.7.5)
Requirement already satisfied: pexpect>4.3 in /Users/madhavsankar/opt/anaconda3/
lib/python3.9/site-packages (from ipython>=7.23.1->ipykernel) (4.8.0)
Requirement already satisfied: jupyter-core>=4.6.0 in /Users/madhavsankar/opt/an
aconda3/lib/python3.9/site-packages (from jupyter-client<8.0->ipykernel) (4.8.1)
Requirement already satisfied: python-dateutil>=2.1 in /Users/madhavsankar/opt/a
naconda3/lib/python3.9/site-packages (from jupyter-client<8.0->ipykernel) (2.8.
2)
Requirement already satisfied: pyzmq>=13 in /Users/madhavsankar/opt/anaconda3/li
b/python3.9/site-packages (from jupyter-client<8.0->ipykernel) (22.2.1)
Requirement already satisfied: parso<0.9.0,>=0.8.0 in /Users/madhavsankar/opt/an
aconda3/lib/python3.9/site-packages (from jedi>=0.16->ipython>=7.23.1->ipykerne
l) (0.8.2)
Requirement already satisfied: ptyprocess>=0.5 in /Users/madhavsankar/opt/anacon
da3/lib/python3.9/site-packages (from pexpect>4.3->ipython>=7.23.1->ipykernel)
(0.7.0)
Requirement already satisfied: wcwidth in /Users/madhavsankar/opt/anaconda3/lib/
python3.9/site-packages (from prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0->ipyt
```

```
hon>=7.23.1->ipykernel) (0.2.5)
Requirement already satisfied: six>=1.5 in /Users/madhavsankar/opt/anaconda3/li
b/python3.9/site-packages (from python-dateutil>=2.1->jupyter-client<8.0->ipyker
nel) (1.16.0)
Requirement already satisfied: scikit-surprise in /Users/madhavsankar/opt/anacon
da3/lib/python3.9/site-packages (1.1.1)
Requirement already satisfied: joblib>=0.11 in /Users/madhavsankar/opt/anaconda
3/lib/python3.9/site-packages (from scikit-surprise) (1.1.0)
Requirement already satisfied: six>=1.10.0 in /Users/madhavsankar/opt/anaconda3/
lib/python3.9/site-packages (from scikit-surprise) (1.16.0)
Requirement already satisfied: numpy>=1.11.2 in /Users/madhavsankar/opt/anaconda
3/lib/python3.9/site-packages (from scikit-surprise) (1.20.3)
Requirement already satisfied: scipy>=1.0.0 in /Users/madhavsankar/opt/anaconda
3/lib/python3.9/site-packages (from scikit-surprise) (1.7.1)
```

```
WARNING:param.Parameterized: Use method 'warning' via param namespace
WARNING:param.main: pandas could not register all extension types imports failed
with the following error: cannot import name 'ABCIndexClass' from 'pandas.core.d
types.generic' (/Users/madhavsankar/opt/anaconda3/lib/python3.9/site-packages/pa
ndas/core/dtypes/generic.py)
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
```

```
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
```

```
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
WARNING:param.Dimension: Use method 'get_param_values' via param namespace
/Users/madhavsankar/opt/anaconda3/lib/python3.9/site-packages/umap/plot.py:66: U
serWarning: Trying to register the cmap 'fire' which already exists.
  plt.register_cmap("fire", fire_cmap)
```

In [36]:

```
import os
current = os.getcwd()
Dataset_loc = current + '/Synthetic_Movie_Lens/'
```

In [37]:

```
data = pd.read_csv(Dataset_loc + "ratings.csv")
data.head()
```

Out[37]:

	userId	movieId	rating	timestamp
0	496	112852	3.0	1415520462
1	391	1947	4.0	1030945141
2	387	1562	1.5	1095041022
3	474	2716	4.5	1053020930
4	483	88125	4.5	1311337237

In [4]:

```
data.describe()
```

Out[4]:

	userId	movieId	rating	timestamp

	userId	movieId	rating	timestamp
count	100836.000000	100836.000000	100836.000000	1.008360e+05
mean	326.127564	19435.295718	3.501562	1.205946e+09
std	182.618491	35530.987199	1.042540	2.162610e+08
min	1.000000	1.000000	0.500000	8.281246e+08
25%	177.000000	1199.000000	3.000000	1.019124e+09
50%	325.000000	2991.000000	3.500000	1.186087e+09
75%	477.000000	8122.000000	4.000000	1.435994e+09
max	610.000000	193609.000000	5.000000	1.537799e+09

In [5]:

```
data = pd.read_csv(Dataset_loc + "movies.csv")
data.head()

data['genres']
data['newgen'] = data['genres'].apply(lambda x: x.split('|'))
newlist = []
for values in data['newgen']:
    newlist += values

print(set(newlist))
print(len(set(newlist))) # 1 of it is 'no genres listed'.
```

```
{'Western', 'Musical', 'Comedy', 'Fantasy', 'Romance', 'Action', 'IMAX', 'Animat
ion', 'Adventure', 'Horror', 'Children', 'Crime', 'Thriller', 'Drama', 'War', 'M
ystery', 'Film-Noir', '(no genres listed)', 'Documentary', 'Sci-Fi'}
20
```

Question 1A

In [6]:

```
Ratings_file = pd.read_csv(Dataset_loc + "ratings.csv",usecols=['userId','movieI
user_ID = Ratings_file.pop('userId').values
movie_ID = Ratings_file.pop('movieId').values
Rating = Ratings_file.pop('rating').values
Sparsity = len(Rating)/(len(set(movie_ID))*len(set(user_ID)))
print('Sparsity:',Sparsity)
```

```
Sparsity: 0.016999683055613623
```

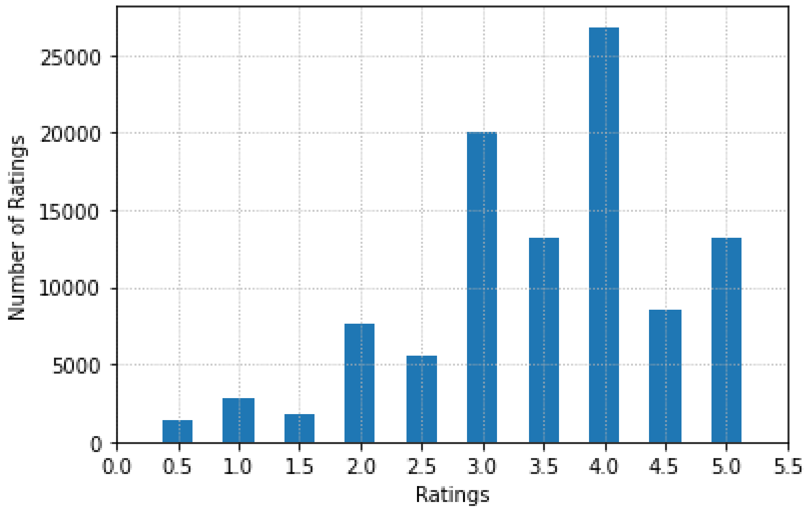
Question 1B

Plot a histogram showing the frequency of the rating values: Bin the raw rating values into intervals of width 0.5 and use the binned rating values as the horizontal axis. Count the number of entries in the ratings matrix R that fall within each bin and use this count as the height of the vertical axis for that particular bin. Comment on the shape of the histogram.

In [7]:

```
uni_values, uni_indices = np.unique(Rating, return_inverse=True)
plt.bar(uni_values, np.bincount(uni_indices), width=0.25)
```

```
locs, labels = plt.xticks()
plt.grid(linestyle=':')
plt.xticks(np.arange(0,6,0.5),rotation=0)
plt.ylabel('Number of Ratings')
plt.xlabel('Ratings')
plt.savefig('Q2_1.png',dpi=500,bbox_inches='tight')
plt.show()
```

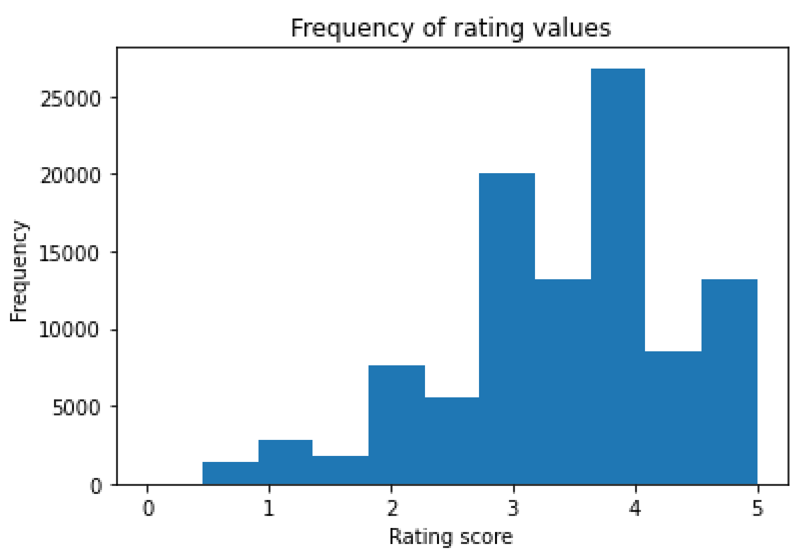


In [23]:

```
bins = np.linspace(0,5,num=12)
plt.hist(Rating,bins=bins)
plt.xlabel("Rating score")
plt.ylabel("Frequency")
plt.title("Frequency of rating values")
```

Out[23]:

```
Text(0.5, 1.0, 'Frequency of rating values')
```

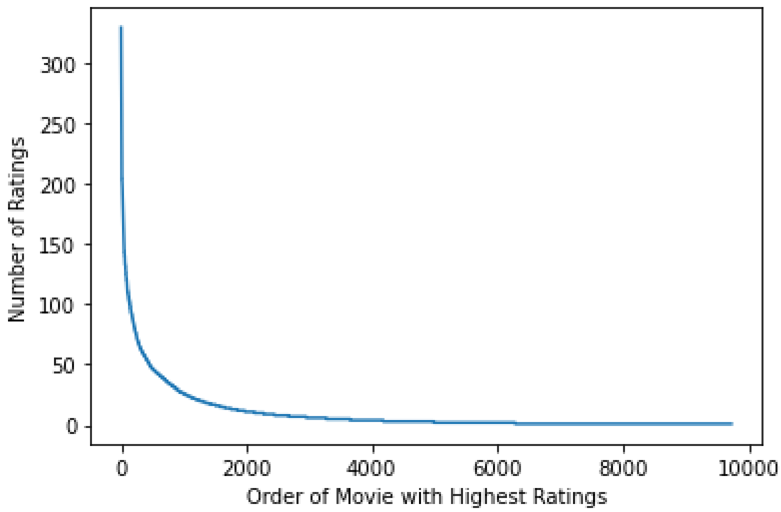


Question 1C

Plot the distribution of the number of ratings received among movies: The X-axis should be the movie index ordered by decreasing frequency and the Y -axis should be the number of ratings the movie has received; ties can broken in any way. A monotonically decreasing trend is expected.

In [13]:
```
#Movie ID vs number of ratings
unique_movie,unique_counts=np.unique(movie_ID,return_counts=True)
count_sorts=np.argsort(unique_counts)
length=range(1,len(unique_movie)+1)
count=unique_counts[count_sorts[::-1]] #Decreasing Frequency
print(count)
plt.plot(length,count)
plt.xlabel('Order of Movie with Highest Ratings')
plt.ylabel('Number of Ratings')
plt.show()
```

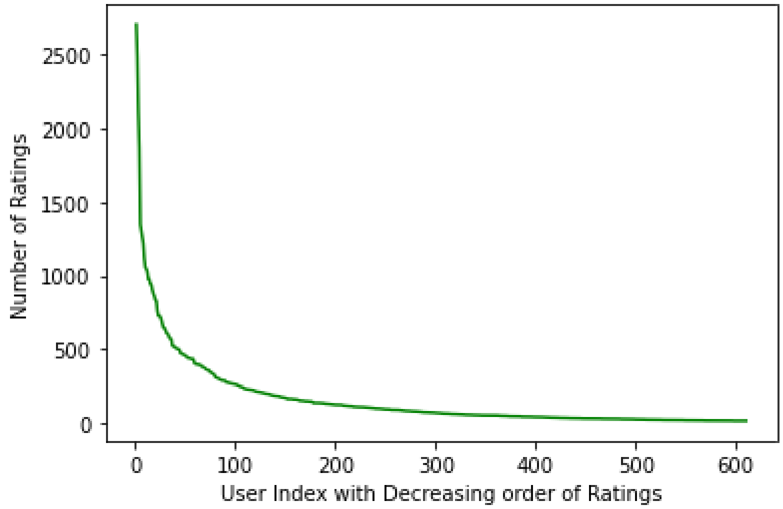
```
[329 317 307 ...   1   1   1]
```



Question 1D

Plot the distribution of ratings among users: The X-axis should be the user index ordered by decreasing frequency and the Y -axis should be the number of movies the user has rated. The requirement of the plot is similar to that in Question C.

In [14]:
```
#User ID vs Number of movies the user has rated
unique_movie1,unique_counts1=np.unique(user_ID,return_counts=True)
count_sorts=np.argsort(unique_counts1)
length1=range(1,len(unique_movie1)+1)
count1=unique_counts1[count_sorts[::-1]] #Decreasing Frequency
#print(count1)
plt.plot(length1,count1,color='g')
plt.xlabel('User Index with Decreasing order of Ratings')
plt.ylabel('Number of Ratings')
plt.show()
```



Question 1F

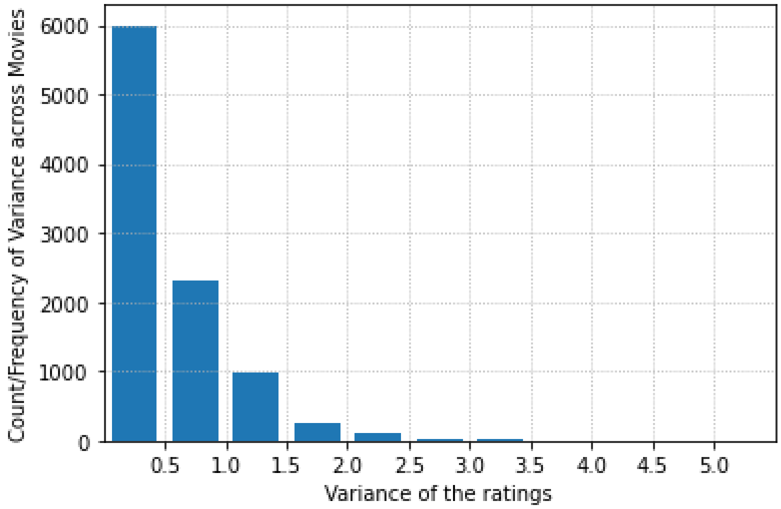
Compute the variance of the rating values received by each movie: Bin the variance values into intervals of width 0.5 and use the binned variance values as the horizontal axis. Count the number of movies with variance values in the binned intervals and use this count as the vertical axis. Briefly comment on the shape of the resulting histogram

In [15]:

```
movie_set=set(movie_ID)
movie_set=list(movie_set) #can't subscript set
list_movies=[]
list_variance=[]
for movie_ele in range(len(movie_set)):
    #For each  unique movie ID, find all it's indexes from the movie ID corpus t
    pos=[ind for ind,ele in enumerate(movie_ID) if ele==movie_set[movie_ele] ]
    #list of all corresponding rattings values to find variance of a single uniq
    variance=np.var(np.array(Rating[pos]))
    list_variance.append(variance)
    list_movies.append(movie_set[movie_ele])
```

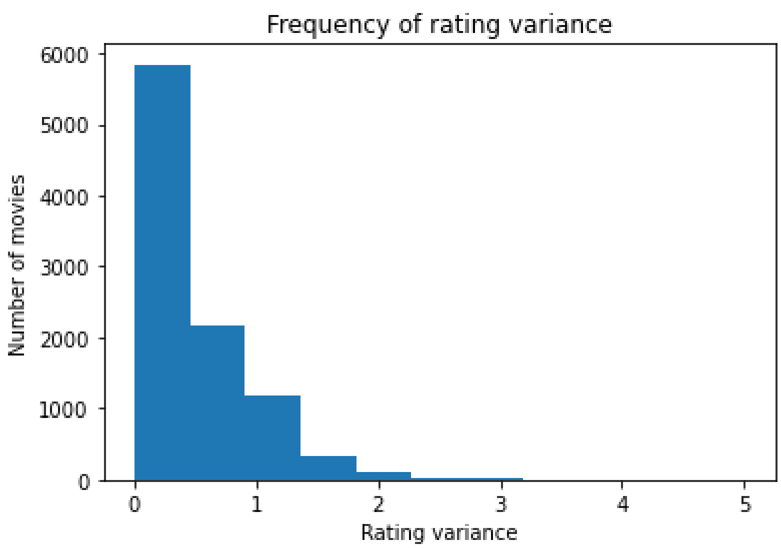
In [25]:

```
#Plotting histogram
plt.hist(list_variance, bins=np.arange(0,5.5,0.5),rwidth=0.75)
plt.xticks(np.arange(0.5,5.5,0.5))
plt.xlim([0, 5.5])
plt.grid(linestyle=':')
plt.xlabel('Variance of the ratings')
plt.ylabel('Count/Frequency of Variance across Movies')
plt.show()
```



In [24]:

```
bins = np.linspace(0,5,num=12)
plt.figure()
plt.hist(list_variance, bins=bins)
plt.xlabel("Rating variance");
plt.ylabel("Number of movies");
plt.title("Frequency of rating variance")
plt.show()
```



Question 2

Understanding the Pearson Correlation Coefficient:

In [19]:

```
print(max(Rating),min(Rating))
```

```
5.0 0.5
```

Question 4

Design a k-NN collaborative filter to predict the ratings of the movies in the original dataset and evaluate its performance using 10-fold cross validation. Sweep k (number of neighbors) from 2 to 100 in step sizes of 2, and for each k compute the average RMSE and average MAE obtained by averaging the RMSE and MAE across all 10 folds. Plot average RMSE (Y-axis) against k (X-axis) and average MAE (Y-axis) against k (X-axis).

In [25]:

```
#Get lineformat corresponding tags CSV file in dataset  - Documentation - https:
CSV_reader = Reader(rating_scale=(0.5, 5),skip_lines=1,sep=',', line_format='use
Dataset_Ratings = Dataset.load_from_file(Dataset_loc+'ratings.csv',reader=CSV_re

k = np.arange(2,102,2)
list_RMSE = []
lisst_mae = []
for ele in k:
    knn = KNNWithMeans(k = ele, sim_options={'name': 'pearson'})
    final_res = cross_validate(knn, measures = ['rmse', 'mae'], data = Dataset_R
    list_RMSE.append(np.mean(final_res['test_rmse']))
    lisst_mae.append(np.mean(final_res['test_mae']))
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

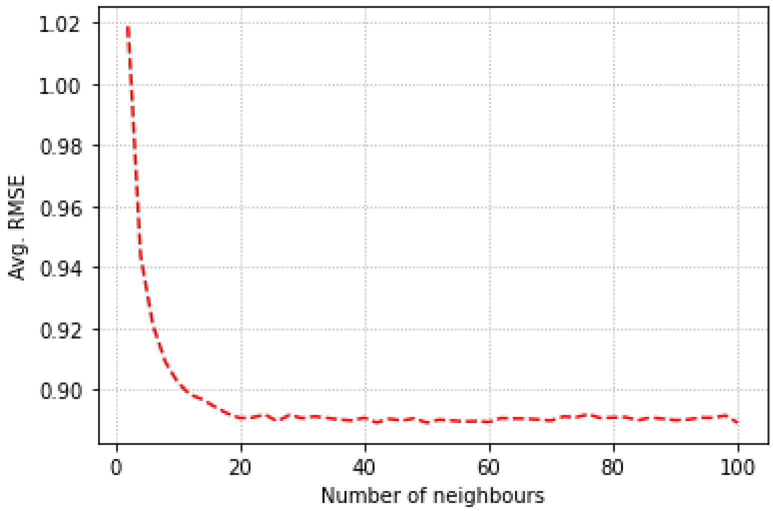
```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

```
Computing the pearson similarity matrix...
Done computing similarity matrix.
```

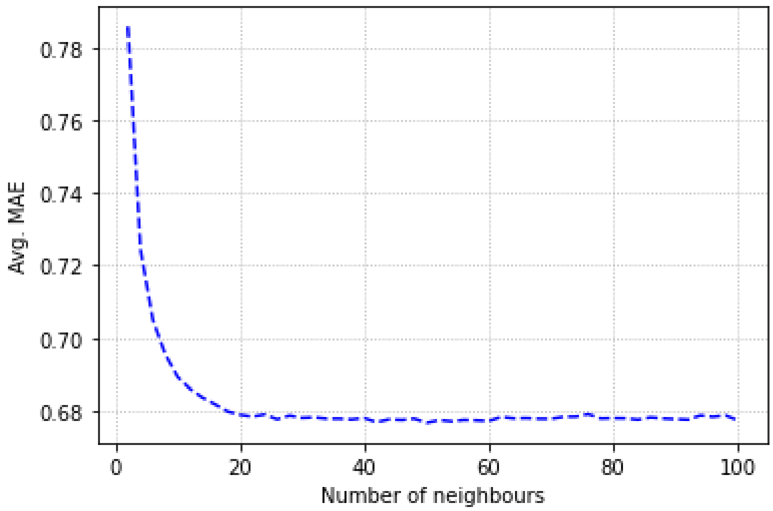
In [27]:

```python
plt.plot(k,list_RMSE,linestyle='--',color='r')
plt.grid(linestyle=':')
plt.ylabel('Avg. RMSE')
plt.xlabel('Number of neighbours')
plt.savefig('Q4a.png',dpi=300,bbox_inches='tight')
plt.show()
```



In [28]:

```python
plt.plot(k,lisst_mae,linestyle='--',color='b')
plt.grid(linestyle=':')
plt.ylabel('Avg. MAE')
plt.xlabel('Number of neighbours')
plt.savefig('Q4b.png',dpi=300,bbox_inches='tight')
plt.show()
```



Question 5

In [26]:

```python
print('k: RMSE, MAE')
for i in range(len(k)):
    print(k[i], ': ', list_RMSE[i], ', ', lisst_mae[i])
```

```
k: RMSE, MAE
2 :  1.018768034893937 ,  0.7857753144380015
4 :  0.9437332859175933 ,  0.7243936934096996
6 :  0.9208146175391345 ,  0.7048910065122361
8 :  0.9088046464861635 ,  0.695423076266558
10 :  0.9020565993711454 ,  0.6891956585451109
12 :  0.8982478646987847 ,  0.6859768704291433
14 :  0.8966523189878487 ,  0.6836034279595953
16 :  0.8942047851489914 ,  0.6817947218490031
18 :  0.8920368254185703 ,  0.6798253236991231
20 :  0.8905586357817494 ,  0.6789263770375126
22 :  0.890734836055375 ,  0.6783961852018932
24 :  0.8917316108129663 ,  0.679008258489914
26 :  0.8895138688417512 ,  0.6776603592356266
28 :  0.8915744018820091 ,  0.6787001868761412
30 :  0.8905046598800069 ,  0.678060219669227
32 :  0.8911121186404991 ,  0.6782140428119852
34 :  0.8906048947884508 ,  0.6778304537595741
36 :  0.890083461983162 ,  0.6778354867918295
38 :  0.889753737176713 ,  0.6776341817446483
40 :  0.8906640752960575 ,  0.6780348061274889
42 :  0.8890709525946228 ,  0.6768498897936197
44 :  0.890300983217897 ,  0.6776518577647475
46 :  0.8897257898188446 ,  0.6774797093194925
48 :  0.8904858863157227 ,  0.6778531049931984
50 :  0.8890178399265855 ,  0.676692385909934
52 :  0.8900314235832468 ,  0.677438299485774
54 :  0.8897300223986153 ,  0.6771153735240285
56 :  0.8894360528502613 ,  0.6774898983921354
58 :  0.8895287046799712 ,  0.6773753789733102
60 :  0.8892595527802968 ,  0.6771876443082434
62 :  0.8905578554548091 ,  0.6783349711401552
64 :  0.8903990679520944 ,  0.6779223616839526
66 :  0.8903494204227039 ,  0.6780073910684403
68 :  0.8900880484702764 ,  0.6778240542741775
70 :  0.889716520195073 ,  0.6777878302375453
72 :  0.8910640260023668 ,  0.6783582430236446
74 :  0.890787505552743 ,  0.6784031823356202
76 :  0.8918108094530046 ,  0.6791049161509691
78 :  0.8905524956644719 ,  0.6779020670542092
80 :  0.8907201772513336 ,  0.6779943704423386
82 :  0.8908664973468152 ,  0.6779482930020613
84 :  0.88980322624066 ,  0.6775688436273838
86 :  0.8907484111532031 ,  0.6781927690270005
88 :  0.8903024707293365 ,  0.6778936651167206
90 :  0.8898495263550185 ,  0.6777185692282449
92 :  0.8900851819626368 ,  0.6775737806527513
94 :  0.8906900801010093 ,  0.6787514735601691
96 :  0.890646302585911 ,  0.6783647800922548
98 :  0.8914162321235171 ,  0.6787889774660207
100 :  0.888998054164175 ,  0.6773526329875889
```

Question 6

In [15]:

```
def pop_trim(data, testset):
    ref = {}
    for j in data.raw_ratings:
```

```
        if j[1] in ref.keys():
            ref[j[1]].append(j[2])
        else:
            ref[j[1]] = []
            ref[j[1]].append(j[2])

    Pop_Trimmed_set = [j for j in testset if len(ref[j[1]]) > 2]
    return Pop_Trimmed_set

def unpop_trim(data, testset):
    ref = {}
    for j in data.raw_ratings:
        if j[1] in ref.keys():
            ref[j[1]].append(j[2])
        else:
            ref[j[1]] = []
            ref[j[1]].append(j[2])

    Unpop_trimmed_set = [j for j in testset if len(ref[j[1]]) <= 2]
    return Unpop_trimmed_set

def highvar_trim(data, testset):
    dict_of_items = {}
    for j in Dataset_Ratings.raw_ratings:
        if j[1] in dict_of_items.keys():
            dict_of_items[j[1]].append(j[2])
        else:
            dict_of_items[j[1]] = []
            dict_of_items[j[1]].append(j[2])

    High_Var_trimmed_set = [j for j in testset if (np.var(dict_of_items[j[1]]) >
    return High_Var_trimmed_set
```

In [86]:

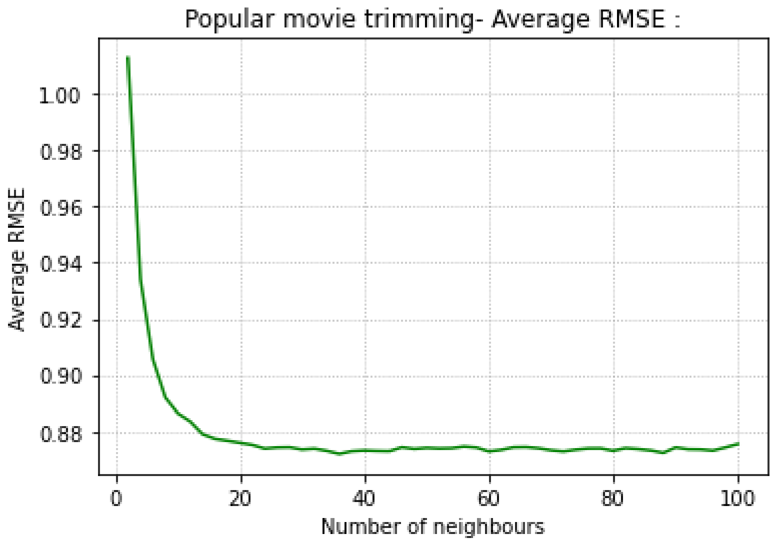
```
from surprise import accuracy

k = np.arange(2,102,2)

Popular_RMSE = []
k_Fold_valid =  KFold(n_splits=10)
for item in k:
    Local_RMSE = []
    print('Running for the iteration with K=',item)
    for trainset, testset in k_Fold_valid.split(Dataset_Ratings):
        Pop_Trimmed_set = pop_trim(Dataset_Ratings, testset)
        res = KNNWithMeans(k=item,sim_options={'name':'pearson'},verbose=False).
        Local_RMSE.append(accuracy.rmse(res,verbose=False))
    Popular_RMSE.append(np.mean(Local_RMSE))
```

```
Running for the iteration with K= 2
Running for the iteration with K= 4
Running for the iteration with K= 6
Running for the iteration with K= 8
Running for the iteration with K= 10
Running for the iteration with K= 12
Running for the iteration with K= 14
Running for the iteration with K= 16
Running for the iteration with K= 18
Running for the iteration with K= 20
```

```
Running for the iteration with K= 22
Running for the iteration with K= 24
Running for the iteration with K= 26
Running for the iteration with K= 28
Running for the iteration with K= 30
Running for the iteration with K= 32
Running for the iteration with K= 34
Running for the iteration with K= 36
Running for the iteration with K= 38
Running for the iteration with K= 40
Running for the iteration with K= 42
Running for the iteration with K= 44
Running for the iteration with K= 46
Running for the iteration with K= 48
Running for the iteration with K= 50
Running for the iteration with K= 52
Running for the iteration with K= 54
Running for the iteration with K= 56
Running for the iteration with K= 58
Running for the iteration with K= 60
Running for the iteration with K= 62
Running for the iteration with K= 64
Running for the iteration with K= 66
Running for the iteration with K= 68
Running for the iteration with K= 70
Running for the iteration with K= 72
Running for the iteration with K= 74
Running for the iteration with K= 76
Running for the iteration with K= 78
Running for the iteration with K= 80
Running for the iteration with K= 82
Running for the iteration with K= 84
Running for the iteration with K= 86
Running for the iteration with K= 88
Running for the iteration with K= 90
Running for the iteration with K= 92
Running for the iteration with K= 94
Running for the iteration with K= 96
Running for the iteration with K= 98
Running for the iteration with K= 100
```

In [87]:

```python
plt.plot(k,Popular_RMSE,color='g')
plt.grid(linestyle=':')
plt.title('Popular movie trimming- Average RMSE :')
plt.ylabel('Average RMSE')
plt.xlabel('Number of neighbours')
plt.savefig('Q6a.png',dpi=300,bbox_inches='tight')
plt.show()
```



In [88]:

```
print("RMSE- Popular movie trimming- Minimum avg. :", min(Popular_RMSE))
print("Value of K: %d" % k[[i for i, x in enumerate(Popular_RMSE) if x == min(Po
```

```
RMSE- Popular movie trimming- Minimum avg. : 0.8719871055017627
Value of K: 36
```

All values are close. So we can use k = 20 as well.

In [89]:

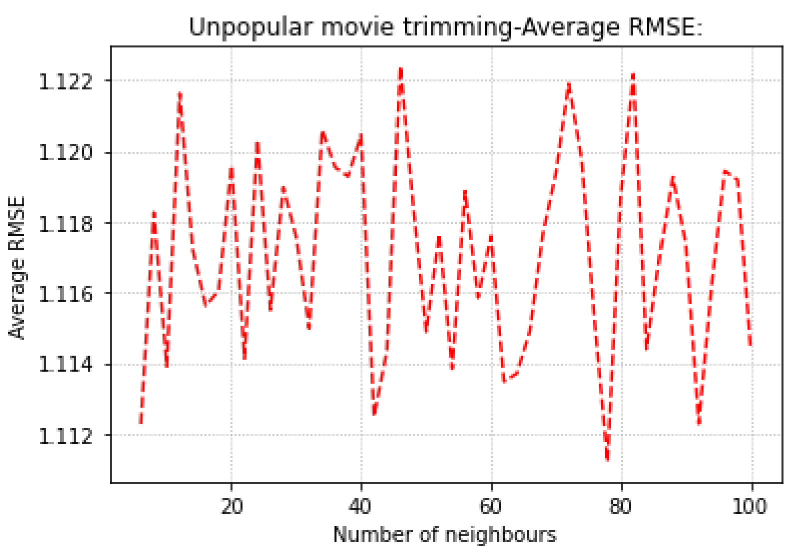
```
UnPopular_RMSE = []
k_Fold_valid =  KFold(n_splits=10)

k = np.arange(2,102,2)

for item in k:
    Local_RMSE = []
    print('Running for the iteration with K=',item)
    for trainset, testset in k_Fold_valid.split(Dataset_Ratings):
        Unpop_trimmed_set = unpop_trim(Dataset_Ratings, testset)
        Unpop_res = KNNWithMeans(k=item,sim_options={'name':'pearson'},verbose=F
        Local_RMSE.append(accuracy.rmse(Unpop_res,verbose=False))
    UnPopular_RMSE.append(np.mean(Local_RMSE))
```

```
Running for the iteration with K= 2
Running for the iteration with K= 4
Running for the iteration with K= 6
Running for the iteration with K= 8
Running for the iteration with K= 10
Running for the iteration with K= 12
Running for the iteration with K= 14
Running for the iteration with K= 16
Running for the iteration with K= 18
Running for the iteration with K= 20
Running for the iteration with K= 22
Running for the iteration with K= 24
Running for the iteration with K= 26
Running for the iteration with K= 28
Running for the iteration with K= 30
Running for the iteration with K= 32
Running for the iteration with K= 34
Running for the iteration with K= 36
```

```
Running for the iteration with K= 38
Running for the iteration with K= 40
Running for the iteration with K= 42
Running for the iteration with K= 44
Running for the iteration with K= 46
Running for the iteration with K= 48
Running for the iteration with K= 50
Running for the iteration with K= 52
Running for the iteration with K= 54
Running for the iteration with K= 56
Running for the iteration with K= 58
Running for the iteration with K= 60
Running for the iteration with K= 62
Running for the iteration with K= 64
Running for the iteration with K= 66
Running for the iteration with K= 68
Running for the iteration with K= 70
Running for the iteration with K= 72
Running for the iteration with K= 74
Running for the iteration with K= 76
Running for the iteration with K= 78
Running for the iteration with K= 80
Running for the iteration with K= 82
Running for the iteration with K= 84
Running for the iteration with K= 86
Running for the iteration with K= 88
Running for the iteration with K= 90
Running for the iteration with K= 92
Running for the iteration with K= 94
Running for the iteration with K= 96
Running for the iteration with K= 98
Running for the iteration with K= 100
```

In [90]:

```python
plt.plot(k[2:],UnPopular_RMSE[2:],linestyle='--',color='r')
plt.grid(linestyle=':')
plt.title('Unpopular movie trimming-Average RMSE:')
plt.ylabel('Average RMSE')
plt.xlabel('Number of neighbours')
plt.savefig('Q13.png',dpi=300,bbox_inches='tight')
plt.show()
```



In [91]:
```
print("Unpopular movie trimming-Minimum average RMSE:", min(UnPopular_RMSE))
print("Value of K: %d" % k[[i for i, x in enumerate(UnPopular_RMSE) if x == min(
```

```
Unpopular movie trimming-Minimum average RMSE: 1.1112045741730798
Value of K: 78
```

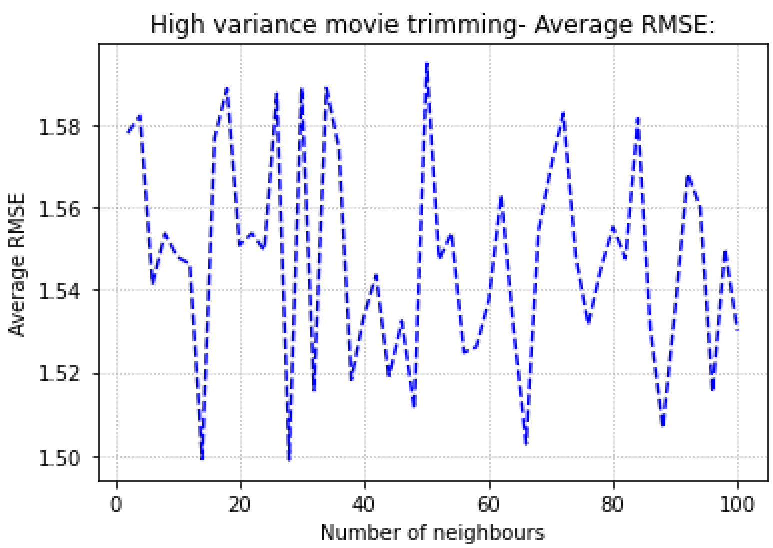
In [71]:
```
#High Variance Movie Trimming
High_Var_RMSE = []
k_Fold_valid =  KFold(n_splits=10)

for item in k:
    Local_RMSE = []
    print('Running for the iteration with K =',item)
    for trainset, testset in k_Fold_valid.split(Dataset_Ratings):
        High_Var_trimmed_set = highvar_trim(Dataset_Ratings, testset)
        Final_High_res = KNNWithMeans(k=item,sim_options={'name':'pearson'},verb
        Local_RMSE.append(accuracy.rmse(Final_High_res,verbose=False))
    High_Var_RMSE.append(np.mean(Local_RMSE))
```

```
Running for the iteration with K = 2
Running for the iteration with K = 4
Running for the iteration with K = 6
Running for the iteration with K = 8
Running for the iteration with K = 10
Running for the iteration with K = 12
Running for the iteration with K = 14
Running for the iteration with K = 16
Running for the iteration with K = 18
Running for the iteration with K = 20
Running for the iteration with K = 22
Running for the iteration with K = 24
Running for the iteration with K = 26
Running for the iteration with K = 28
Running for the iteration with K = 30
Running for the iteration with K = 32
Running for the iteration with K = 34
Running for the iteration with K = 36
Running for the iteration with K = 38
Running for the iteration with K = 40
Running for the iteration with K = 42
Running for the iteration with K = 44
Running for the iteration with K = 46
Running for the iteration with K = 48
Running for the iteration with K = 50
Running for the iteration with K = 52
Running for the iteration with K = 54
Running for the iteration with K = 56
Running for the iteration with K = 58
Running for the iteration with K = 60
Running for the iteration with K = 62
Running for the iteration with K = 64
Running for the iteration with K = 66
Running for the iteration with K = 68
Running for the iteration with K = 70
Running for the iteration with K = 72
Running for the iteration with K = 74
Running for the iteration with K = 76
Running for the iteration with K = 78
Running for the iteration with K = 80
```

```
Running for the iteration with K = 82
Running for the iteration with K = 84
Running for the iteration with K = 86
Running for the iteration with K = 88
Running for the iteration with K = 90
Running for the iteration with K = 92
Running for the iteration with K = 94
Running for the iteration with K = 96
Running for the iteration with K = 98
Running for the iteration with K = 100
```

In [72]:

```
plt.plot(k,High_Var_RMSE,linestyle='--',color='b')
plt.grid(linestyle=':')
plt.title('High variance movie trimming- Average RMSE:')
plt.ylabel('Average RMSE')
plt.xlabel('Number of neighbours')
plt.savefig('Q14.png',dpi=300,bbox_inches='tight')
plt.show()
```



In [73]:

```
print("High variance movie trimming- Minimum average RMSE:", min(High_Var_RMSE))
print("Value of K: %d" % k[[i for i, x in enumerate(High_Var_RMSE) if x == min(H
```

```
High variance movie trimming- Minimum average RMSE: 1.499004534919353
Value of K: 28
```

In [ ]:

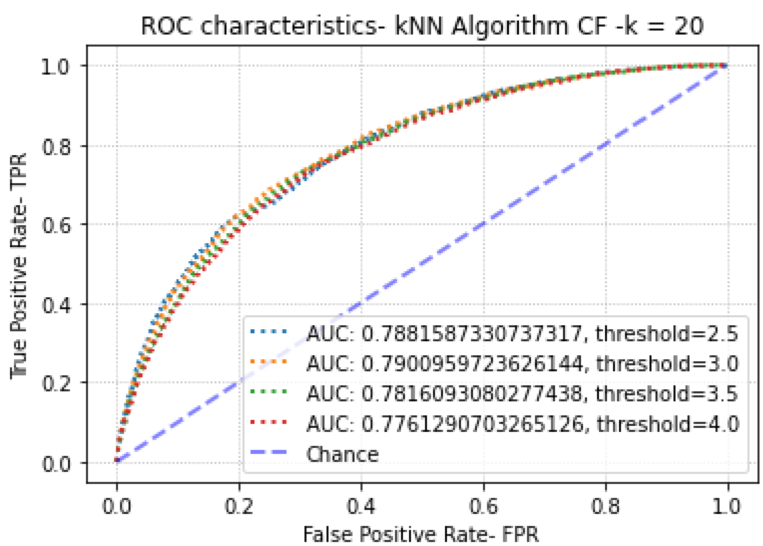
```
CSV_reader=Reader(rating_scale=(0.5, 5),skip_lines=1,sep=',', line_format='user
Dataset_Ratings=Dataset.load_from_file(Dataset_loc+'ratings.csv',reader=CSV_read

k = 20
Train_list, Test_list = train_test_split(Dataset_Ratings, test_size=0.1)
Thres_list = [2.5, 3.0, 3.5, 4.0]

res  = KNNWithMeans(k=k,sim_options={'name':'pearson'},verbose=False).fit(Train_
```

```
Requirement already satisfied: scikit-surprise in c:\users\veera\appdata\local\p
rograms\python\python39\lib\site-packages (1.1.1)
Requirement already satisfied: scipy>=1.0.0 in c:\users\veera\appdata\local\prog
rams\python\python39\lib\site-packages (from scikit-surprise) (1.7.3)
Requirement already satisfied: joblib>=0.11 in c:\users\veera\appdata\local\prog
rams\python\python39\lib\site-packages (from scikit-surprise) (1.1.0)
```

```
Requirement already satisfied: numpy>=1.11.2 in c:\users\veera\appdata\local\pro
grams\python\python39\lib\site-packages (from scikit-surprise) (1.21.5)
Requirement already satisfied: six>=1.10.0 in c:\users\veera\appdata\local\progr
ams\python\python39\lib\site-packages (from scikit-surprise) (1.15.0)
WARNING: You are using pip version 21.2.4; however, version 22.0.3 is available.
You should consider upgrading via the 'c:\users\veera\appdata\local\programs\pyt
hon\python39\python.exe -m pip install --upgrade pip' command.
```

In [ ]:

```python
fig, ax = plt.subplots()
for item in Thres_list:
    thresholded_out = []
    for row in res:
        if row.r_ui > item:
            thresholded_out.append(1)
        else:
            thresholded_out.append(0)
    FPR, TPR, thresholds = roc_curve(thresholded_out, [row.est for row in res])
    ax.plot(FPR, TPR,lw=2,linestyle=':',label="AUC: "+str(auc(FPR,TPR))+', thres
ax.plot([0, 1], [0, 1], linestyle='--', lw=2, color='b', label='Chance', alpha=.
plt.legend(loc='best')
plt.grid(linestyle=':')
plt.title('ROC characteristics- kNN Algorithm CF -k = 20')
plt.ylabel('True Positive Rate- TPR')
plt.xlabel('False Positive Rate- FPR')
plt.savefig('Q6.png',dpi=350,bbox_inches='tight')
plt.show()
```

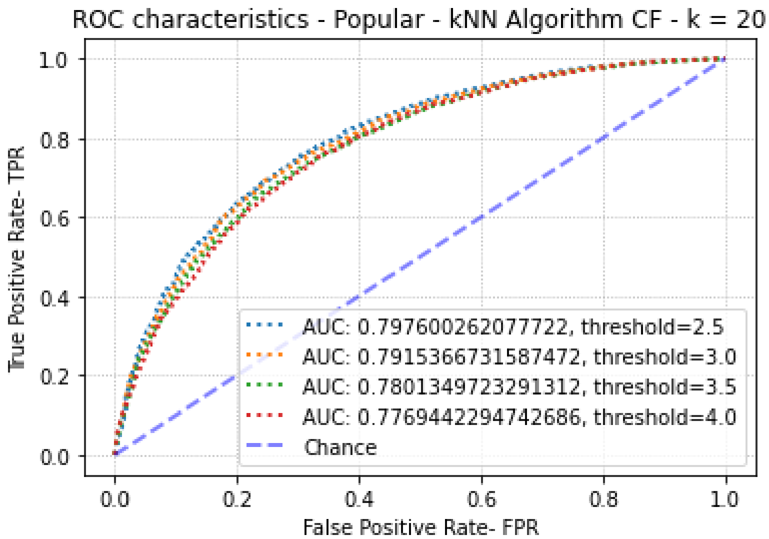


Popular

In [92]:

```python
CSV_reader=Reader(rating_scale=(0.5, 5),skip_lines=1,sep=',', line_format='user
Dataset_Ratings=Dataset.load_from_file(Dataset_loc+'ratings.csv',reader=CSV_read

k = 20
Train_list, Test_list = train_test_split(Dataset_Ratings, test_size=0.1)
Thres_list = [2.5, 3.0, 3.5, 4.0]

Pop_Trimmed_set = pop_trim(Dataset_Ratings, Test_list)
res  = KNNWithMeans(k=k,sim_options={'name':'pearson'},verbose=False).fit(Train_

fig, ax = plt.subplots()
```

```
for item in Thres_list:
    thresholded_out = []
    for row in res:
        if row.r_ui > item:
            thresholded_out.append(1)
        else:
            thresholded_out.append(0)
    FPR, TPR, thresholds = roc_curve(thresholded_out, [row.est for row in res])
    ax.plot(FPR, TPR,lw=2,linestyle=':',label="AUC: "+str(auc(FPR,TPR))+', thres
ax.plot([0, 1], [0, 1], linestyle='--', lw=2, color='b', label='Chance', alpha=.
plt.legend(loc='best')
plt.grid(linestyle=':')
plt.title('ROC characteristics - Popular - kNN Algorithm CF - k = 20')
plt.ylabel('True Positive Rate- TPR')
plt.xlabel('False Positive Rate- FPR')
plt.savefig('Q6a.png',dpi=350,bbox_inches='tight')
plt.show()
```

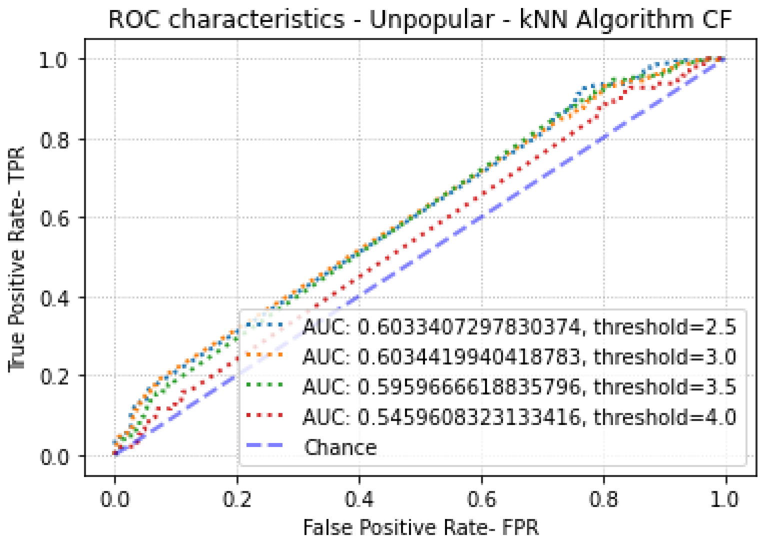


Unpopular

In [99]:

```
CSV_reader=Reader(rating_scale=(0.5, 5),skip_lines=1,sep=',', line_format='user
Dataset_Ratings=Dataset.load_from_file(Dataset_loc+'ratings.csv',reader=CSV_read

k = 20
Train_list, Test_list = train_test_split(Dataset_Ratings, test_size=0.1)
Thres_list = [2.5, 3.0, 3.5, 4.0]

Unpop_trimmed_set = unpop_trim(Dataset_Ratings, Test_list)

res  = KNNWithMeans(k=k,sim_options={'name':'pearson'},verbose=False).fit(Train_

fig, ax = plt.subplots()
for item in Thres_list:
    thresholded_out = []
    for row in res:
        if row.r_ui > item:
            thresholded_out.append(1)
        else:
            thresholded_out.append(0)
    FPR, TPR, thresholds = roc_curve(thresholded_out, [row.est for row in res])
    ax.plot(FPR, TPR,lw=2,linestyle=':',label="AUC: "+str(auc(FPR,TPR))+', thres
```

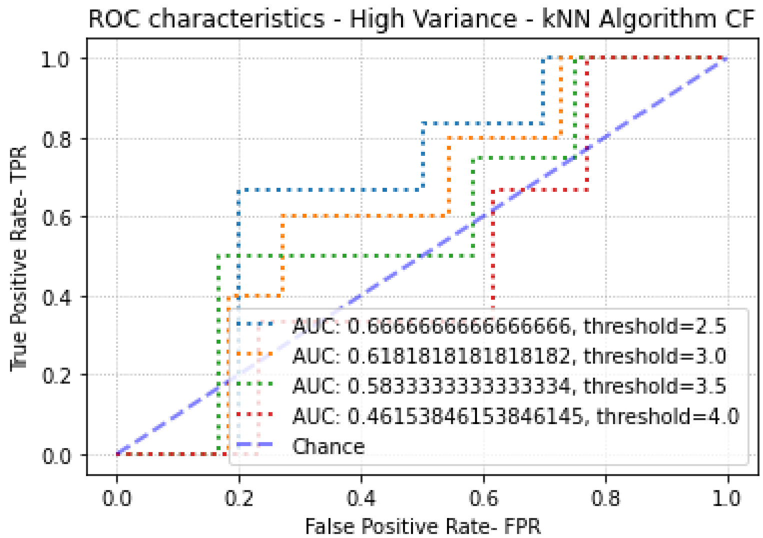
```
ax.plot([0, 1], [0, 1], linestyle='--', lw=2, color='b', label='Chance', alpha=.
plt.legend(loc='best')
plt.grid(linestyle=':')
plt.title('ROC characteristics - Unpopular - kNN Algorithm CF')
plt.ylabel('True Positive Rate- TPR')
plt.xlabel('False Positive Rate- FPR')
plt.savefig('Q6b.png',dpi=350,bbox_inches='tight')
plt.show()
```



High Variance

In [101…

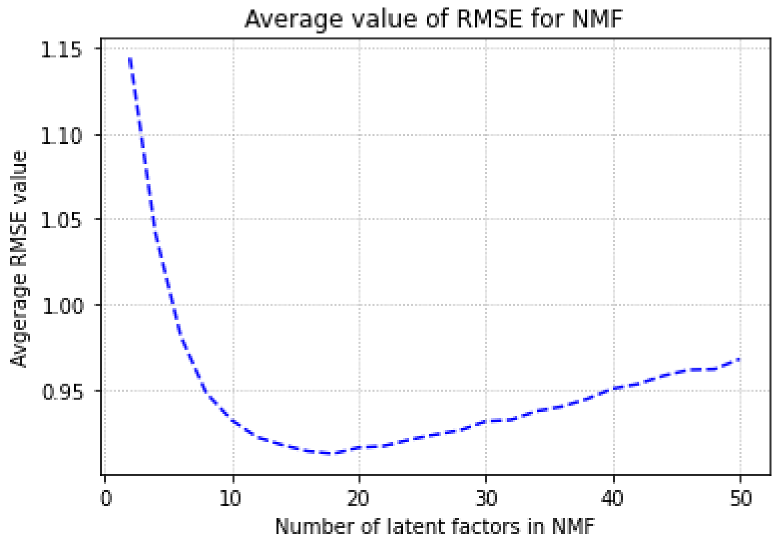
```
CSV_reader=Reader(rating_scale=(0.5, 5),skip_lines=1,sep=',', line_format='user
Dataset_Ratings=Dataset.load_from_file(Dataset_loc+'ratings.csv',reader=CSV_read

k = 20
Train_list, Test_list = train_test_split(Dataset_Ratings, test_size=0.1)
Thres_list = [2.5, 3.0, 3.5, 4.0]

High_Var_trimmed_set = highvar_trim(Dataset_Ratings, Test_list)
res  = KNNWithMeans(k=k,sim_options={'name':'pearson'},verbose=False).fit(Train_

fig, ax = plt.subplots()
for item in Thres_list:
    thresholded_out = []
    for row in res:
        if row.r_ui > item:
            thresholded_out.append(1)
        else:
            thresholded_out.append(0)
    FPR, TPR, thresholds = roc_curve(thresholded_out, [row.est for row in res])
    ax.plot(FPR, TPR,lw=2,linestyle=':',label="AUC: "+str(auc(FPR,TPR))+', thres
ax.plot([0, 1], [0, 1], linestyle='--', lw=2, color='b', label='Chance', alpha=.
plt.legend(loc='best')
plt.grid(linestyle=':')
plt.title('ROC characteristics - High Variance - kNN Algorithm CF')
plt.ylabel('True Positive Rate- TPR')
plt.xlabel('False Positive Rate- FPR')
plt.savefig('Q6c.png',dpi=350,bbox_inches='tight')
plt.show()
```



Question 8A

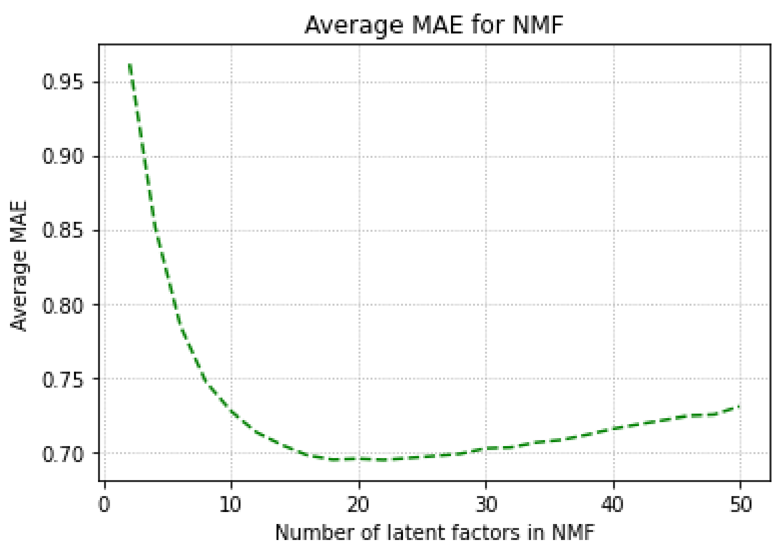
In [ ]:

```
from surprise.prediction_algorithms.matrix_factorization import NMF, SVD

List_K = np.arange(2,52,2)
RMSE_NMF_List = []
MAE_NMF_List = []
for K_value in List_K:
    res = cross_validate(NMF(n_factors=K_value,n_epochs=50,verbose=False),
                         measures=['rmse','mae'],data = Dataset_Ratings,cv=10,n_
    RMSE_NMF_List.append(np.mean(res['test_rmse']))
    MAE_NMF_List.append(np.mean(res['test_mae']))

plt.plot(List_K,RMSE_NMF_List,linestyle='--',color='b')
plt.grid(linestyle=':')
plt.title('Average value of RMSE for NMF')
plt.ylabel('Avgerage RMSE value')
plt.xlabel('Number of latent factors in NMF')
plt.savefig('Q8a.png',dpi=350,bbox_inches='tight')
plt.show()

plt.plot(List_K,RMSE_NMF_List,linestyle='--',color='b')
plt.grid(linestyle=':')
plt.title('Average value of RMSE for NMF')
plt.ylabel('Avgerage RMSE value')
plt.xlabel('Number of latent factors in NMF')
plt.savefig('Q8a.png',dpi=350,bbox_inches='tight')
plt.show()
```



In [ ]:

```
plt.plot(List_K,MAE_NMF_List,linestyle='--',color='g')
plt.grid(linestyle=':')
plt.title('Average MAE for NMF')
plt.ylabel('Average MAE')
plt.xlabel('Number of latent factors in NMF')
plt.savefig('Q8b.png',dpi=350,bbox_inches='tight')
plt.show()
```



Question 8B

In [ ]:

```
print("Minimum Average value of RMSE (NMF): %f, value of K: %d" % (min(RMSE_NMF_
print("Minimum Average value of MAE (NMF): %f, value of K: %d" % (min(MAE_NMF_Li
```

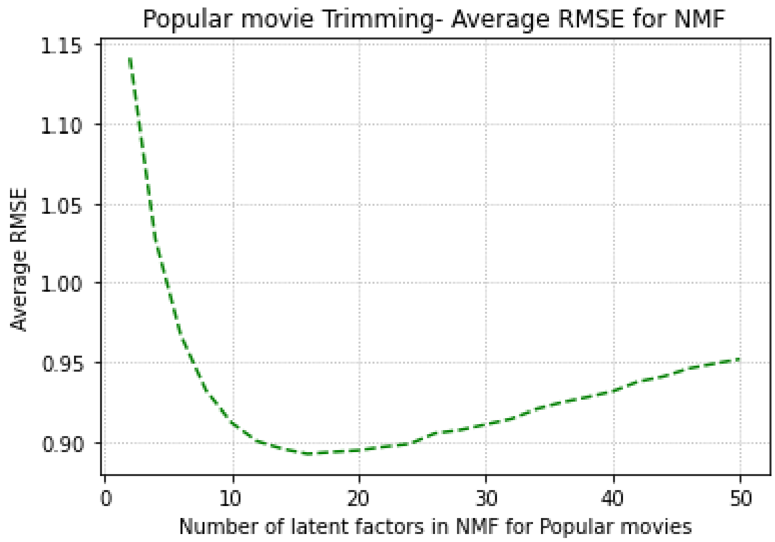
```
Minimum Average value of RMSE (NMF): 0.912530, value of K: 18
Minimum Average value of MAE (NMF): 0.694350, value of K: 22
```

Question 8C

In [97]:

```
Pop_RMSE_NMF_list = []
K_Fold =  KFold(n_splits=10)

for K_value in List_K:
    Local_RMSE = []
    print('Iterating for the K value =',K_value)
    for Train_list, Test_list in K_Fold.split(Dataset_Ratings):
        trimmed_set = pop_trim(Dataset_Ratings, Test_list)
        res = NMF(n_factors=K_value,n_epochs=50,verbose=False).fit(Train_list).t
        Local_RMSE.append(accuracy.rmse(res,verbose=False))
    Pop_RMSE_NMF_list.append(np.mean(Local_RMSE))
plt.plot(List_K,Pop_RMSE_NMF_list,linestyle='--',color='g')
plt.grid(linestyle=':')
plt.title('Popular movie Trimming- Average RMSE for NMF')
plt.ylabel('Average RMSE')
plt.xlabel('Number of latent factors in NMF for Popular movies')
plt.savefig('Q8c.png',dpi=350,bbox_inches='tight')
plt.show()
```

```
Iterating for the K value = 2
Iterating for the K value = 4
Iterating for the K value = 6
Iterating for the K value = 8
Iterating for the K value = 10
Iterating for the K value = 12
Iterating for the K value = 14
Iterating for the K value = 16
Iterating for the K value = 18
Iterating for the K value = 20
Iterating for the K value = 22
Iterating for the K value = 24
Iterating for the K value = 26
Iterating for the K value = 28
Iterating for the K value = 30
Iterating for the K value = 32
Iterating for the K value = 34
Iterating for the K value = 36
Iterating for the K value = 38
Iterating for the K value = 40
Iterating for the K value = 42
Iterating for the K value = 44
Iterating for the K value = 46
Iterating for the K value = 48
Iterating for the K value = 50
```



In [102…

```
print("Minimum avg. RMSE (NMF, Popular movie trimming):", min(Pop_RMSE_NMF_list)
```

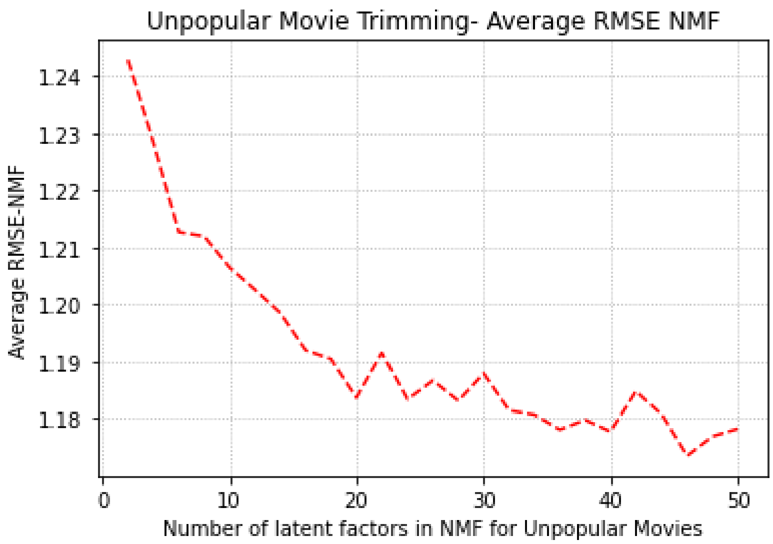
```
Minimum avg. RMSE (NMF, Popular movie trimming): 0.8925320415716778
```

In [103…

```
UnPop_RMSE_NMF = []
k_Fold =  KFold(n_splits=10)
for K_Value in List_K:
    Local_RMSE = []
    print('Iteration for the value of K =',K_Value)
    for Train_list, Test_list in k_Fold.split(Dataset_Ratings):
        trimmed_set = unpop_trim(Dataset_Ratings, Test_list)
        res = NMF(n_factors=K_Value,n_epochs=50,verbose=False).fit(Train_list).t
        Local_RMSE.append(accuracy.rmse(res,verbose=False))
    UnPop_RMSE_NMF.append(np.mean(Local_RMSE))
```

```
Iteration for the value of K = 2
Iteration for the value of K = 4
Iteration for the value of K = 6
Iteration for the value of K = 8
Iteration for the value of K = 10
Iteration for the value of K = 12
Iteration for the value of K = 14
Iteration for the value of K = 16
Iteration for the value of K = 18
Iteration for the value of K = 20
Iteration for the value of K = 22
Iteration for the value of K = 24
Iteration for the value of K = 26
Iteration for the value of K = 28
Iteration for the value of K = 30
Iteration for the value of K = 32
Iteration for the value of K = 34
Iteration for the value of K = 36
Iteration for the value of K = 38
Iteration for the value of K = 40
Iteration for the value of K = 42
Iteration for the value of K = 44
Iteration for the value of K = 46
Iteration for the value of K = 48
Iteration for the value of K = 50
```

In [104…
```
plt.plot(List_K,UnPop_RMSE_NMF,linestyle='--',color='r')
plt.grid(linestyle=':')
plt.title('Unpopular Movie Trimming- Average RMSE NMF')
plt.ylabel('Average RMSE-NMF')
plt.xlabel('Number of latent factors in NMF for Unpopular Movies')
plt.savefig('Q8c2.png',dpi=350,bbox_inches='tight')
plt.show()
```



In [105…
```
print(" Unpopular movie Trimming- Minimum average RMSE -NMF", min(UnPop_RMSE_NMF
```

```
 Unpopular movie Trimming- Minimum average RMSE -NMF 1.173498893400049
```

In [16]:
```
Var_RMSE_NMF = []
k_Fold =  KFold(n_splits=10)

List_K = np.arange(2,52,2)
CSV_reader=Reader(rating_scale=(0.5, 5),skip_lines=1,sep=',', line_format='user
Dataset_Ratings=Dataset.load_from_file(Dataset_loc+'ratings.csv',reader=CSV_read
for K_value in List_K:
    Local_RMSE = []
    print('Iteration for the value of K =',K_value)
    for Train_list, Test_list in k_Fold.split(Dataset_Ratings):
        trimmed_set = highvar_trim(Dataset_Ratings, Test_list)
        res = NMF(n_factors=K_value,n_epochs=50,verbose=False).fit(Train_list).t
        Local_RMSE.append(accuracy.rmse(res,verbose=False))
    Var_RMSE_NMF.append(np.mean(Local_RMSE))
```

```
Iteration for the value of K = 2
Iteration for the value of K = 4
Iteration for the value of K = 6
Iteration for the value of K = 8
Iteration for the value of K = 10
Iteration for the value of K = 12
Iteration for the value of K = 14
Iteration for the value of K = 16
Iteration for the value of K = 18
Iteration for the value of K = 20
Iteration for the value of K = 22
Iteration for the value of K = 24
Iteration for the value of K = 26
```

```
Iteration for the value of K = 28
Iteration for the value of K = 30
Iteration for the value of K = 32
Iteration for the value of K = 34
Iteration for the value of K = 36
Iteration for the value of K = 38
Iteration for the value of K = 40
Iteration for the value of K = 42
Iteration for the value of K = 44
Iteration for the value of K = 46
Iteration for the value of K = 48
Iteration for the value of K = 50
```

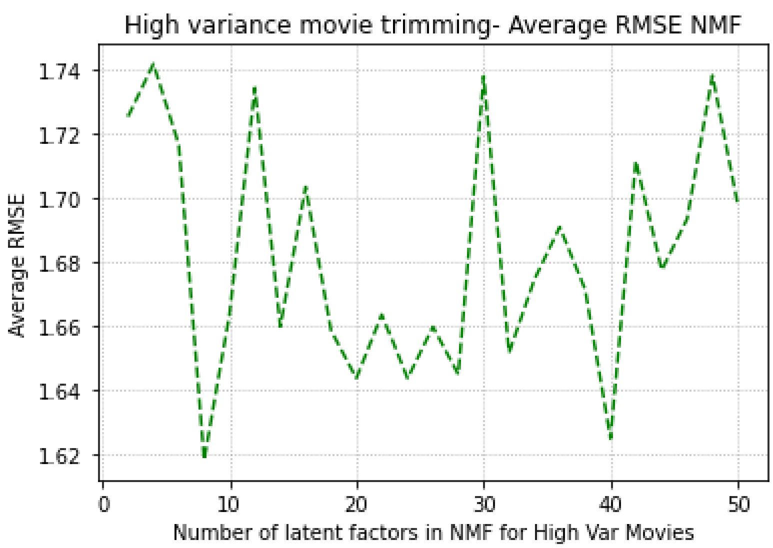
In [17]:

```
print("High Variance movie Trimming- Minimum average RMSE -NMF", min(Var_RMSE_NM
```

```
High Variance movie Trimming- Minimum average RMSE -NMF 1.6182565408745588
```

In [18]:

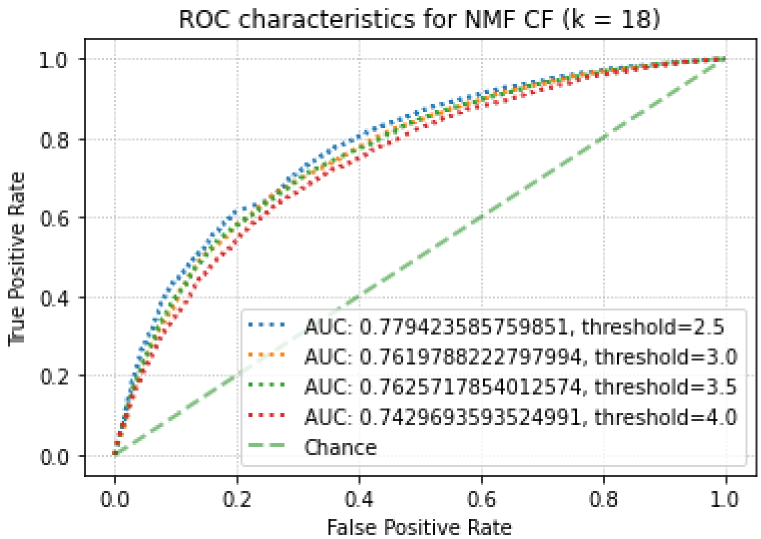
```
plt.plot(List_K,Var_RMSE_NMF,linestyle='--',color='g')
plt.grid(linestyle=':')
plt.title('High variance movie trimming- Average RMSE NMF')
plt.ylabel('Average RMSE')
plt.xlabel('Number of latent factors in NMF for High Var Movies')
plt.savefig('Q8c3.png',dpi=350,bbox_inches='tight')
plt.show()
```



In [ ]:

```
k = List_K[[i for i, x in enumerate(RMSE_NMF_List) if x == min(RMSE_NMF_List)][0
Thresh_List = [2.5, 3.0, 3.5, 4.0]
Train_list, Test_list = train_test_split(Dataset_Ratings, test_size=0.1)
res  = NMF(n_factors=k,n_epochs=50,verbose=False).fit(Train_list).test(Test_list

fig, ax = plt.subplots()
for item in Thresh_List:
    thresholded_out = []
    for row in res:
        if row.r_ui > item:
            thresholded_out.append(1)
        else:
            thresholded_out.append(0)
    fpr, tpr, thresholds = roc_curve(thresholded_out, [row.est for row in res])
    ax.plot(fpr, tpr,lw=2,linestyle=':',label="AUC: "+str(auc(fpr,tpr))+', thres
```

```
ax.plot([0, 1], [0, 1], linestyle='--', lw=2, color='g', label='Chance', alpha=.
plt.legend(loc='best')
plt.grid(linestyle=':')
plt.title('ROC characteristics for NMF CF (k = 18)')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.savefig('Q8d.png',dpi=350,bbox_inches='tight')
plt.show()
```



Question 9

In [ ]:

```
genre = pd.read_csv(Dataset_loc+'movies.csv',usecols=['movieId','title','genres'
Train_list, Test_list = train_test_split(Dataset_Ratings, test_size=0.1)
NMF_K20 = NMF(n_factors=20,n_epochs=50,verbose=False)
NMF_K20.fit(Train_list).test(Test_list)
U_mat = NMF_K20.pu
V_mat = NMF_K20.qi
```

In [24]:

```
cols = [1,3,5,7,11,15,19]
for item in cols:
    print('Column number correspoding to V Matrix Instance: ',item)
    selected_col = V_mat[:,item]
    sorted_col = np.argsort(selected_col)[::-1]
    newlist = []
    for i in sorted_col[0:10]:
        print(genre['genres'][i])
        lst = genre['genres'][i].split('|')
        newlist += lst

    print('Genres: ', set(newlist))
    print('Number of Unique Genres:', len(set(newlist)))

    print('%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%')
```

```
Column number correspoding to V Matrix Instance:  1
Adventure
Horror|Mystery|Thriller
Children|Comedy
```

```
Documentary
Crime|Drama|Mystery|Thriller
Drama
Action|Adventure|Sci-Fi|IMAX
Comedy
Drama|Western
Comedy|Sci-Fi
Genres:  {'Documentary', 'Sci-Fi', 'Adventure', 'Comedy', 'Crime', 'Action', 'Ch
ildren', 'Drama', 'Thriller', 'Horror', 'Western', 'Mystery', 'IMAX'}
Number of Unique Genres: 13
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Column number correspoding to V Matrix Instance:  3
Action|Children|Sci-Fi|IMAX
Action|Drama|Sci-Fi
Horror
Action|Drama|War
Children
Drama
Horror|Mystery|Thriller
Comedy
Mystery|Thriller
Comedy|Drama
Genres:  {'Sci-Fi', 'Comedy', 'War', 'Action', 'Children', 'Drama', 'Thriller',
'Horror', 'Mystery', 'IMAX'}
Number of Unique Genres: 10
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Column number correspoding to V Matrix Instance:  5
Drama|Horror
Action|Crime|Drama
Comedy|Crime
Adventure|Drama|Romance
Adventure|Drama|Thriller
Action|Comedy
Comedy
Comedy|Musical|Romance
Horror
Drama|Romance
Genres:  {'Adventure', 'Comedy', 'Crime', 'Action', 'Drama', 'Musical', 'Romanc
e', 'Thriller', 'Horror'}
Number of Unique Genres: 9
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Column number correspoding to V Matrix Instance:  7
Comedy
Comedy|Drama
Sci-Fi
Comedy
Crime|Drama
Documentary
Thriller
Comedy|Fantasy
Comedy
Adventure|Comedy|Drama|Fantasy|Romance
Genres:  {'Documentary', 'Sci-Fi', 'Adventure', 'Comedy', 'Crime', 'Drama', 'Rom
ance', 'Thriller', 'Fantasy'}
Number of Unique Genres: 9
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Column number correspoding to V Matrix Instance:  11
Action|Horror|Thriller
Documentary
Adventure|Drama|Western
```

```
Action|Crime|Thriller
Comedy
Drama|Western
Drama
Action|Drama|Horror
Action|Comedy
Adventure|Comedy|Crime
Genres:  {'Documentary', 'Adventure', 'Comedy', 'Crime', 'Action', 'Drama', 'Thr
iller', 'Horror', 'Western'}
Number of Unique Genres: 9
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Column number correspoding to V Matrix Instance:  15
Action
Drama
Children|Fantasy
Drama|Fantasy|Romance
Adventure|Comedy|Fantasy
Comedy
Action|Adventure|Drama|Western
Adventure|Animation|Comedy
Comedy|Romance
Comedy|Drama
Genres:  {'Adventure', 'Comedy', 'Action', 'Children', 'Drama', 'Romance', 'West
ern', 'Animation', 'Fantasy'}
Number of Unique Genres: 9
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Column number correspoding to V Matrix Instance:  19
Crime|Drama|Thriller
Adventure|Drama|Romance
Horror
Comedy|Musical|Romance
Comedy
Drama
Comedy|Horror|Romance
Drama|Horror|Sci-Fi|Thriller
Western
Crime|Thriller
Genres:  {'Sci-Fi', 'Adventure', 'Comedy', 'Crime', 'Drama', 'Musical', 'Romanc
e', 'Thriller', 'Horror', 'Western'}
Number of Unique Genres: 10
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

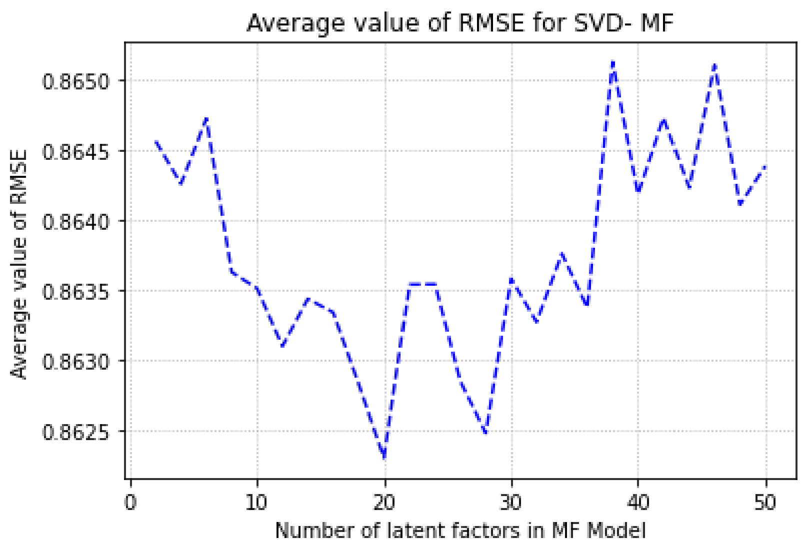
Question 10A

In [ ]:

```
#MF Collaborative Filter
#Treat MF with bias/regularization as SVD kind of optimization problem

List_K = np.arange(2,52,2)
RMSE_MF_SVD = []
MAE_MF_SVD = []
for item in List_K:
    res = cross_validate(SVD(n_factors=item, n_epochs=30,verbose=False),
                         measures=['rmse','mae'],data = Dataset_Ratings, cv=10,
    RMSE_MF_SVD.append(np.mean(res['test_rmse']))
    MAE_MF_SVD.append(np.mean(res['test_mae']))
```

In [54]:

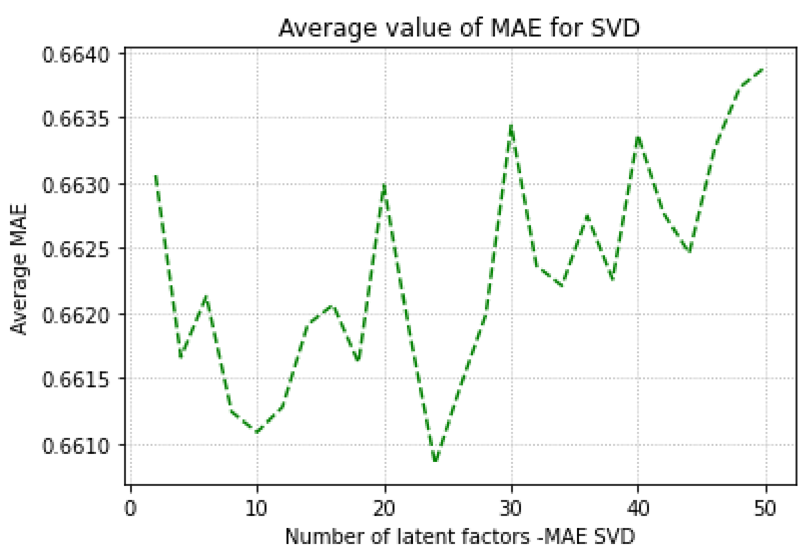
```
plt.plot(List_K,RMSE_MF_SVD,linestyle='--',color='b')
plt.grid(linestyle=':')
plt.title('Average value of RMSE for SVD- MF')
plt.ylabel('Average value of RMSE')
plt.xlabel('Number of latent factors in MF Model')
plt.savefig('Q10a.png',dpi=350,bbox_inches='tight')
plt.show()

print("Minimum average value of RMSE for SVD: %f, K Value: %d" % (min(RMSE_MF_SV
```




```
Minimum average value of RMSE for SVD: 0.862305, K Value: 20
```

In [48]:

```
plt.plot(List_K,MAE_MF_SVD,linestyle='--',color='g')
plt.grid(linestyle=':')
plt.title('Average value of MAE for SVD')
plt.ylabel('Average MAE')
plt.xlabel('Number of latent factors -MAE SVD')
plt.savefig('Q10a2.png',dpi=350,bbox_inches='tight')
plt.show()
```



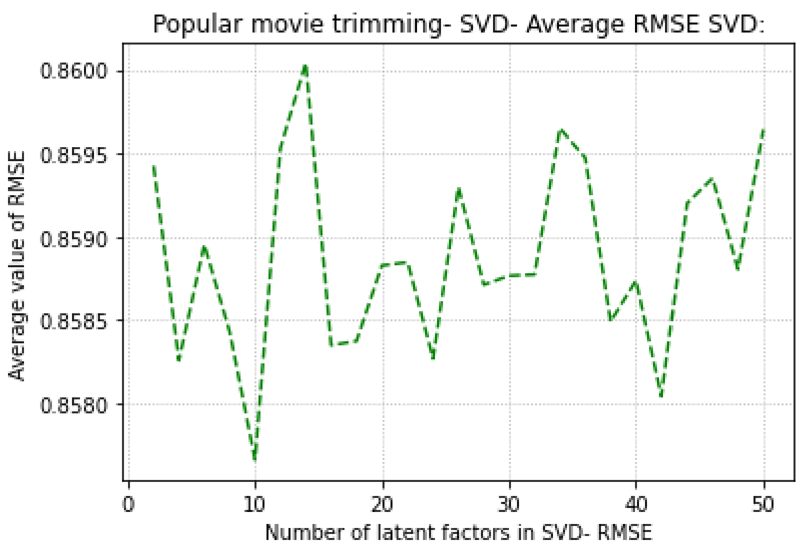
Question 10B

In [50]:
```
print("Minimum average value of MAE for SVD: %f, K Value: %d" % (min(MAE_MF_SVD)
```

```
Minimum average value of MAE for SVD: 0.660847, K Value: 24
```

Question 10C

In [106…
```
Popular_RMSE_SVD = []
k_fold =  KFold(n_splits=10)
for item in List_K:
    Local_RMSE = []
    for Train_list, Test_list in k_fold.split(Dataset_Ratings):
        trimmed_set = pop_trim(Dataset_Ratings, Test_list)
        res = SVD(n_factors=item,n_epochs=20,verbose=False).fit(Train_list).test
        Local_RMSE.append(accuracy.rmse(res,verbose=False))
    Popular_RMSE_SVD.append(np.mean(Local_RMSE))

plt.plot(List_K,Popular_RMSE_SVD,linestyle='--',color='g')
plt.grid(linestyle=':')
plt.title('Popular movie trimming- SVD- Average RMSE SVD:')
plt.ylabel('Average value of RMSE')
plt.xlabel('Number of latent factors in SVD- RMSE')
plt.savefig('Q10c.png',dpi=300,bbox_inches='tight')
plt.show()
```



In [107…
```
print("Popular movie trimming- Minimum average value of RMSE- SVD", min(Popular_
```

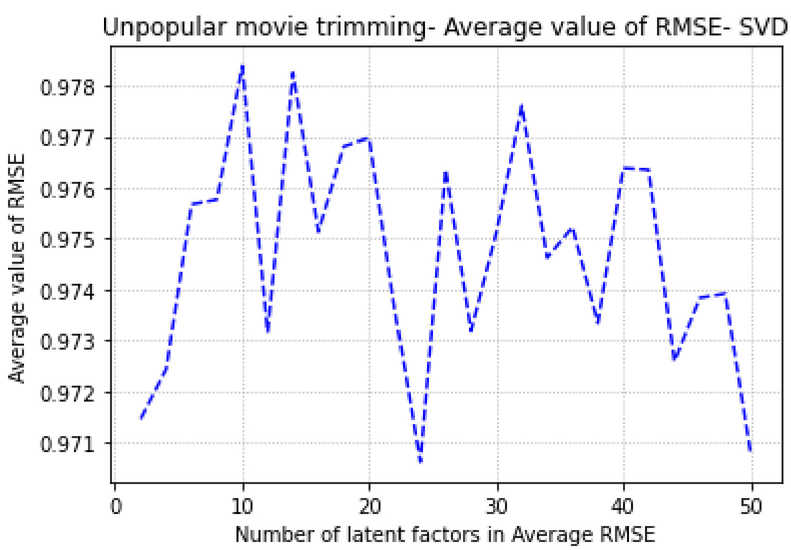
```
Popular movie trimming- Minimum average value of RMSE- SVD 0.8576636998948253
```

In [108…
```
Unpop_RMSE_SVD = []
k_fold =  KFold(n_splits=10)
for item in List_K:
    Local_RMSE = []
    print('Iteration for the value of K =',item)
    for Train_list, Test_list in k_fold.split(Dataset_Ratings):
        trimmed_set = unpop_trim(Dataset_Ratings, Test_list)
        res = SVD(n_factors=item,n_epochs=20,verbose=False).fit(Train_list).test
        Local_RMSE.append(accuracy.rmse(res,verbose=False))
    Unpop_RMSE_SVD.append(np.mean(Local_RMSE))
```

```
Iteration for the value of K = 2
Iteration for the value of K = 4
Iteration for the value of K = 6
Iteration for the value of K = 8
Iteration for the value of K = 10
Iteration for the value of K = 12
Iteration for the value of K = 14
Iteration for the value of K = 16
Iteration for the value of K = 18
Iteration for the value of K = 20
Iteration for the value of K = 22
Iteration for the value of K = 24
Iteration for the value of K = 26
Iteration for the value of K = 28
Iteration for the value of K = 30
Iteration for the value of K = 32
Iteration for the value of K = 34
Iteration for the value of K = 36
Iteration for the value of K = 38
Iteration for the value of K = 40
Iteration for the value of K = 42
Iteration for the value of K = 44
Iteration for the value of K = 46
Iteration for the value of K = 48
Iteration for the value of K = 50
```

In [109…

```
plt.plot(List_K,Unpop_RMSE_SVD,linestyle='--',color='b')
plt.grid(linestyle=':')
plt.title('Unpopular movie trimming- Average value of RMSE- SVD')
plt.ylabel('Average value of RMSE')
plt.xlabel('Number of latent factors in Average RMSE')
plt.savefig('Q10c2.png',dpi=350,bbox_inches='tight')
plt.show()
```



In [110…

```
print("Unpopular movie trimming- Minimum average value of RMSE -SVD):", min(Unpo
```

```
Unpopular movie trimming- Minimum average value of RMSE -SVD): 0.970618365193595
9
```

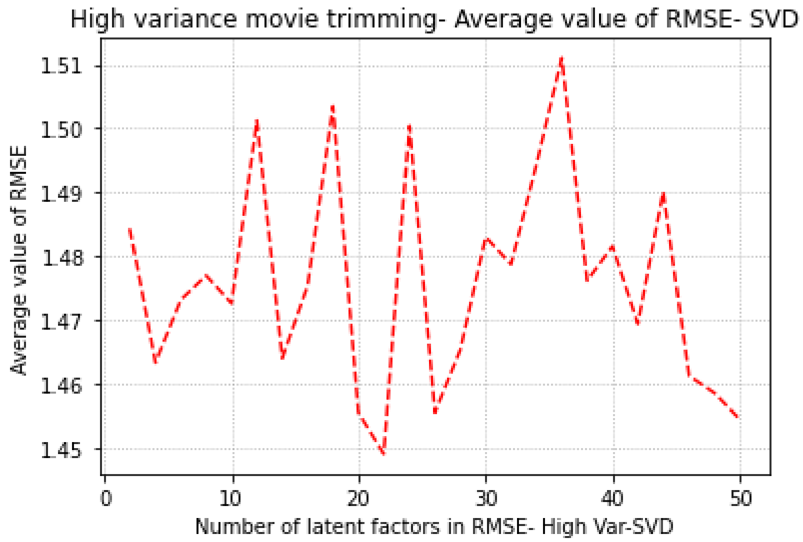
In [ ]:

```
High_Var_RMSE_SVD = []
k_fold =  KFold(n_splits=10)

for item in List_K:
    Local_RMSE = []
    print('Iteration for the value of K =',item)
    for Train_list, Test_list in k_fold.split(Dataset_Ratings):
        Var_trim_set = highvar_trim(Dataset_Ratings, Test_list)
        res = SVD(n_factors=item,n_epochs=20,verbose=False).fit(Train_list).test
        Local_RMSE.append(accuracy.rmse(res,verbose=False))
    High_Var_RMSE_SVD.append(np.mean(Local_RMSE))
```

```
Iteration for the value of K = 2
Iteration for the value of K = 4
Iteration for the value of K = 6
Iteration for the value of K = 8
Iteration for the value of K = 10
Iteration for the value of K = 12
Iteration for the value of K = 14
Iteration for the value of K = 16
Iteration for the value of K = 18
Iteration for the value of K = 20
Iteration for the value of K = 22
Iteration for the value of K = 24
Iteration for the value of K = 26
Iteration for the value of K = 28
Iteration for the value of K = 30
Iteration for the value of K = 32
Iteration for the value of K = 34
Iteration for the value of K = 36
Iteration for the value of K = 38
Iteration for the value of K = 40
Iteration for the value of K = 42
Iteration for the value of K = 44
Iteration for the value of K = 46
Iteration for the value of K = 48
Iteration for the value of K = 50
```

In [ ]:

```
plt.plot(List_K,High_Var_RMSE_SVD,linestyle='--',color='r')
plt.grid(linestyle=':')
plt.title('High variance movie trimming- Average value of RMSE- SVD')
plt.ylabel('Average value of RMSE')
plt.xlabel('Number of latent factors in RMSE- High Var-SVD')
plt.savefig('Q10c3.png',dpi=300,bbox_inches='tight')
plt.show()
```



In [ ]:

```
print("High variance movie trimming- Minimum average value of RMSE- SVD", min(Hi
```

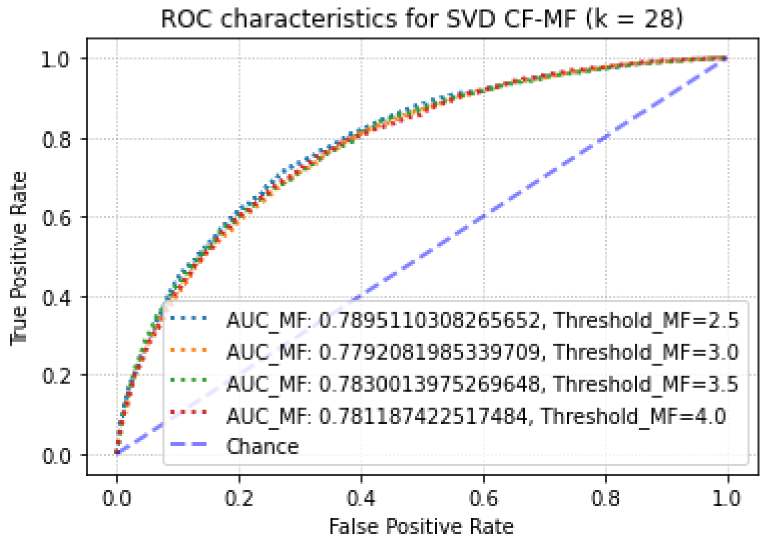
```
High variance movie trimming- Minimum average value of RMSE- SVD 1.4489840651169
374
```

In [ ]:

```
# Use K's value found in Question b to find ROC curves and associated areas

K_Value = List_K[[i for i, x in enumerate(RMSE_MF_SVD) if x == min(RMSE_MF_SVD)]
thres = [2.5, 3.0, 3.5, 4.0]
Train_list, Test_list = train_test_split(Dataset_Ratings, test_size=0.1)
res  = SVD(n_factors=K_Value,n_epochs=20,verbose=False).fit(Train_list).test(Tes
```

In [ ]:

```
fig, ax = plt.subplots()
for thres_val in thres:
    thresholded_out = []
    for row in res:
        if row.r_ui > thres_val:
            thresholded_out.append(1)
        else:
            thresholded_out.append(0)
    FPR_MF, TPR_MF, thresholds = roc_curve(thresholded_out, [row.est for row in
    ax.plot(FPR_MF, TPR_MF,lw=2,linestyle=':',label="AUC_MF: "+str(auc(FPR_MF,TP
ax.plot([0, 1], [0, 1], linestyle='--', lw=2, color='b', label='Chance', alpha=.
plt.legend(loc='best')
plt.grid(linestyle=':')
plt.title('ROC characteristics for SVD CF-MF (k = '+ str(K_Value)+')')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.savefig('Q10d.png',dpi=350,bbox_inches='tight')
plt.show()
```



Question 11

In [112…

```
Set_User_ID = list(set(user_ID))
Mean_user_Ratings = []
for User_idx in Set_User_ID:
    idx = np.where(user_ID == User_idx)
    Mean_user_Ratings.append(np.mean(Rating[idx]))
```

In [ ]:

```
k_fold = KFold(n_splits=10)
Local_RMSE = []
for Train_list, Test_list in k_fold.split(Dataset_Ratings):
    res = [Mean_user_Ratings[int(row[0])-1] for row in Test_list]
    gt = [row[2] for row in Test_list]
    Local_RMSE.append(mean_squared_error(gt,res,squared=False))
Naive_CF_RMSE = np.mean(Local_RMSE)
print('Average RMSE value for Naive Collaborative Filtering: ',Naive_CF_RMSE)
```

```
Average RMSE value for Naive Collaborative Filtering:  0.934689278702231
```

In [113…

```
Popular_Naive_CF_Local_RMSE = []
k_fold =  KFold(n_splits=10)
for Train_list, Test_list in k_fold.split(Dataset_Ratings):
    trimmed_set = pop_trim(Dataset_Ratings, Test_list)
    res = [Mean_user_Ratings[int(row[0])-1] for row in trimmed_set]
    gt = [row[2] for row in trimmed_set]
    Popular_Naive_CF_Local_RMSE.append(mean_squared_error(gt,res,squared=False))
Pop_Naive_CF_RMSE = np.mean(Popular_Naive_CF_Local_RMSE)
print('Popular movie trimming- Average RMSE value for Naive Filtering: ',Pop_Nai
```

```
Popular movie trimming- Average RMSE value for Naive Filtering:  0.9323153270544
065
```

In [114…

```
UnPopular_Naive_CF_Local_RMSE = []
k_fold =  KFold(n_splits=10)
for Train_list, Test_list in k_fold.split(Dataset_Ratings):
    trimmed_set = unpop_trim(Dataset_Ratings, Test_list)
    res = [Mean_user_Ratings[int(row[0])-1] for row in trimmed_set]
```

```
    gt = [row[2] for row in trimmed_set]
    UnPopular_Naive_CF_Local_RMSE.append(mean_squared_error(gt,res,squared=False
UnPop_Naive_CF_RMSE = np.mean(UnPopular_Naive_CF_Local_RMSE)
print('Unpopular movie trimming- Average RMSE value for Naive Filtering: ',UnPop
```

```
Unpopular movie trimming- Average RMSE value for Naive Filtering:  0.97107255400
0641
```

In [ ]:

```
High_Var_Naive_CF_Local_RMSE = []
k_fold =  KFold(n_splits=10)

for Train_list, Test_list in k_fold.split(Dataset_Ratings):
    trimmed_set = highvar_trim(Dataset_Ratings, Test_list)
    res = [Mean_user_Ratings[int(row[0])-1] for row in trimmed_set]
    gt = [row[2] for row in trimmed_set]
    High_Var_Naive_CF_Local_RMSE.append(mean_squared_error(gt,res,squared=False)
High_Var_Naive_CF_RMSE = np.mean(High_Var_Naive_CF_Local_RMSE)
print('High Variance movie trimming- Average RMSE value for Naive Filtering: ',H
```

```
High Variance movie trimming- Average RMSE value for Naive Filtering:  1.4816032
71265438
```

Question 12

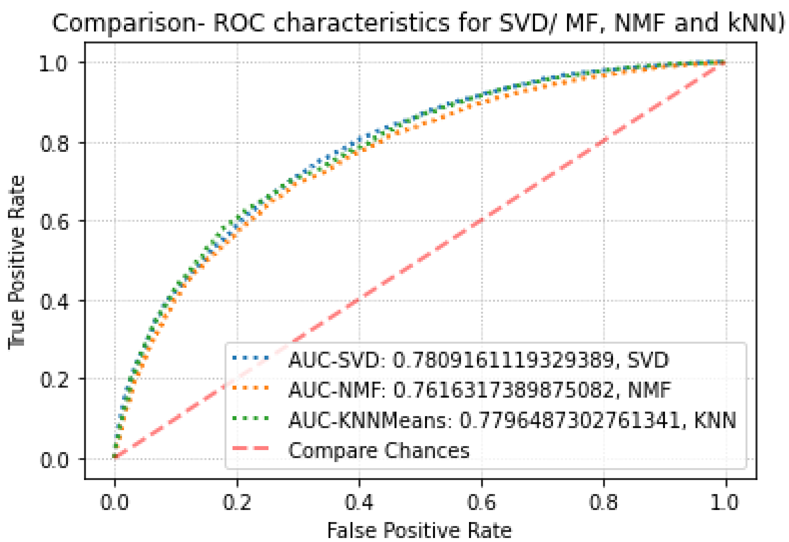
In [ ]:

```
Train_list, Test_list = train_test_split(Dataset_Ratings, test_size=0.1)
SVD_Result  = SVD(n_factors=20,n_epochs=20,verbose=False).fit(Train_list).test(T
NMF_Result  = NMF(n_factors=18,n_epochs=50,verbose=False).fit(Train_list).test(T
KNNMeans_Result = KNNWithMeans(k=20,sim_options={'name':'pearson'},verbose=False
```

In [ ]:

```
fig, ax = plt.subplots()
Thresholded_Result = []
for row in SVD_Result:
    if row.r_ui > 3:
        Thresholded_Result.append(1)
    else:
        Thresholded_Result.append(0)
FPR_SVD, TPR_SVD, thresholds = roc_curve(Thresholded_Result, [row.est for row in
ax.plot(FPR_SVD, TPR_SVD,lw=2,linestyle=':',label="AUC-SVD: "+str(auc(FPR_SVD,TP

Thresholded_Result = []
for row in NMF_Result:
    if row.r_ui > 3:
        Thresholded_Result.append(1)
    else:
        Thresholded_Result.append(0)
FPR_NMF, TPR_NMF, thresholds = roc_curve(Thresholded_Result, [row.est for row in
ax.plot(FPR_NMF, TPR_NMF,lw=2,linestyle=':',label="AUC-NMF: "+str(auc(FPR_NMF,TP

Thresholded_Result = []
for row in KNNMeans_Result:
    if row.r_ui > 3:
        Thresholded_Result.append(1)
    else:
        Thresholded_Result.append(0)
FPR_KNNMeans, TPR_KNNMeans, thresholds = roc_curve(Thresholded_Result, [row.est
ax.plot(FPR_KNNMeans, TPR_KNNMeans,lw=2,linestyle=':',label="AUC-KNNMeans: "+str
```

```
ax.plot([0, 1], [0, 1], linestyle='--', lw=2, color='r', label='Compare Chances'
plt.legend(loc='best')
plt.grid(linestyle=':')
plt.title('Comparison- ROC characteristics for SVD/ MF, NMF and kNN)')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.savefig('Q12.png',dpi=350,bbox_inches='tight')
plt.show()
```



Question 14

In [ ]:

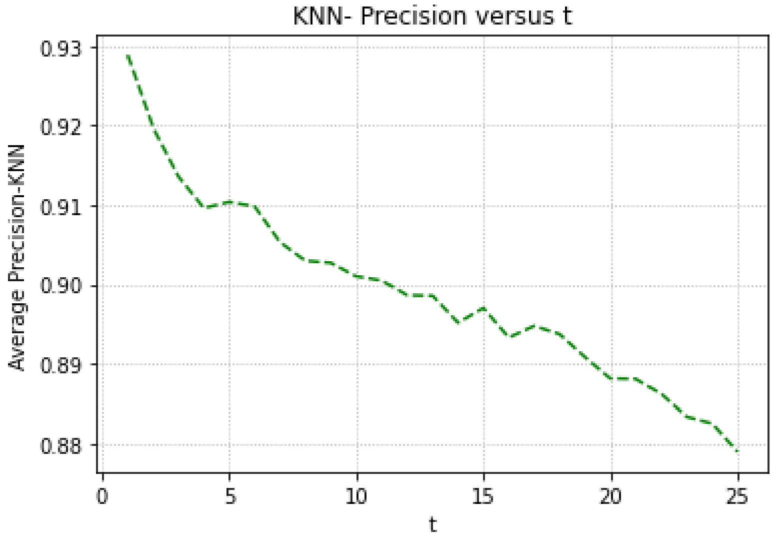
```
#Precision Recall Metrics for Different Models
# K here are the t values not 'k', which is set to 20.
K = np.arange(1,26,1)
k_fold = KFold(n_splits=10)

prec_list_KNN = []
rec_list_KNN = []
for val in K:
    print('Iterating  for the value of K =',val)
    precision_local_set = []
    recall__local_set = []
    for Train_list, Test_list in k_fold.split(Dataset_Ratings):
        D = {} #dictionary of movies liked by users
        for row in Test_list:
            if row[0] in D.keys():
                if row[2] >= 3.0:
                    D[row[0]].add(row[1])
            else:
                D[row[0]] = set()
                if row[2] >= 3.0:
                    D[row[0]].add(row[1])
        dict_of_items = {} #dictionary of all movies rated by users
        for row in Test_list:
            if row[0] in dict_of_items.keys():
                dict_of_items[row[0]].append(row[1])
            else:
                dict_of_items[row[0]] = []
```

```
                dict_of_items[row[0]].append(row[1])
        KNN_Mod_Testlist = [row for row in Test_list if (len(dict_of_items[row[0
        res = KNNWithMeans(k=20,sim_options={'name':'pearson'},verbose=False).fi
        Est_Ratings = {} #dictionary of estimated ratings by users
        for row in res:
            if row[0] in Est_Ratings.keys():
                Est_Ratings[row[0]].append((row[1],row[3]))
            else:
                Est_Ratings[row[0]] = []
                Est_Ratings[row[0]].append((row[1],row[3]))
        precision_u = []
        recall_u = []
        for item in Est_Ratings.keys():
            Set_all = Est_Ratings[item]
            Set_all = sorted(Set_all,key=lambda x:x[1],reverse=True)
            Set_K = set([row[0] for row in Set_all[0:val]])
            precision_u.append(len(Set_K.intersection(D[item]))/float(len(Set_K)
            recall_u.append(len(Set_K.intersection(D[item]))/float(len(D[item]))
        precision_local_set.append(np.mean(precision_u))
        recall__local_set.append(np.mean(recall_u))
    prec_list_KNN.append(np.mean(precision_local_set))
    rec_list_KNN.append(np.mean(recall__local_set))
```

```
Iterating  for the value of K = 1
Iterating  for the value of K = 2
Iterating  for the value of K = 3
Iterating  for the value of K = 4
Iterating  for the value of K = 5
Iterating  for the value of K = 6
Iterating  for the value of K = 7
Iterating  for the value of K = 8
Iterating  for the value of K = 9
Iterating  for the value of K = 10
Iterating  for the value of K = 11
Iterating  for the value of K = 12
Iterating  for the value of K = 13
Iterating  for the value of K = 14
Iterating  for the value of K = 15
Iterating  for the value of K = 16
Iterating  for the value of K = 17
Iterating  for the value of K = 18
Iterating  for the value of K = 19
Iterating  for the value of K = 20
Iterating  for the value of K = 21
Iterating  for the value of K = 22
Iterating  for the value of K = 23
Iterating  for the value of K = 24
Iterating  for the value of K = 25
```

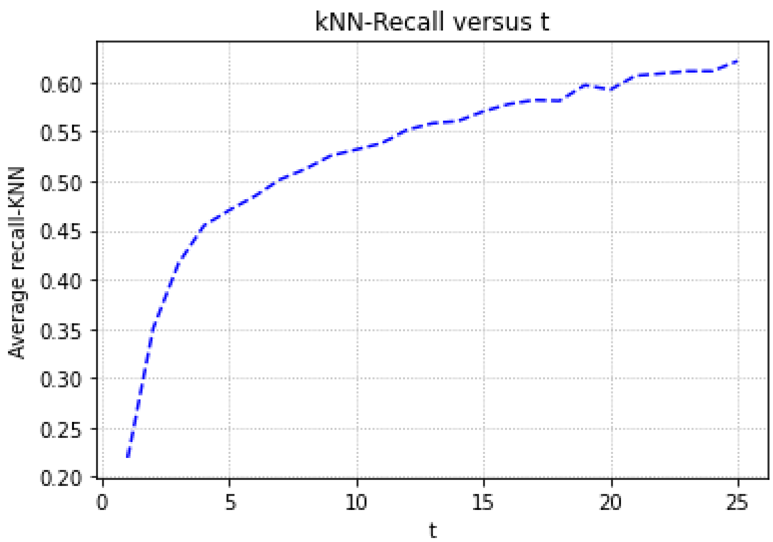
In [ ]:

```
plt.plot(K,prec_list_KNN,linestyle='--',color='g')
plt.grid(linestyle=':')
plt.title('KNN- Precision versus t')
plt.ylabel('Average Precision-KNN')
plt.xlabel('t')
plt.savefig('Q14a1.png',dpi=350,bbox_inches='tight')
plt.show()
```



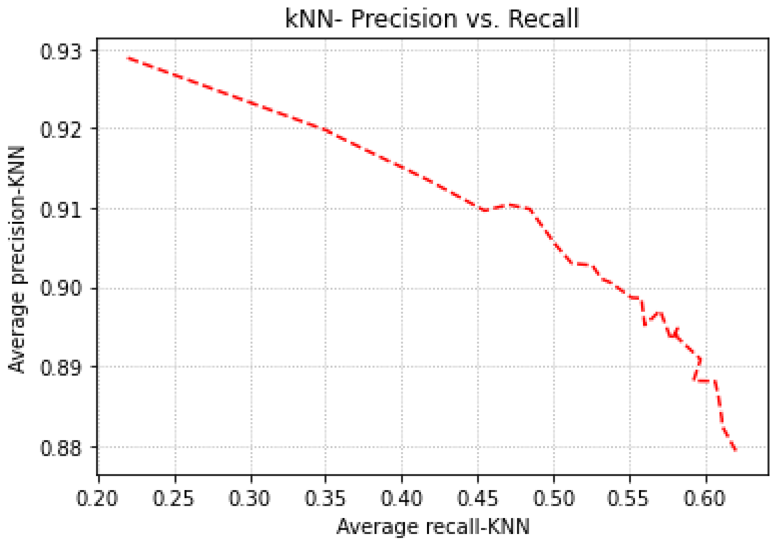
In [ ]:

```
plt.plot(K,rec_list_KNN,linestyle='--',color='b')
plt.grid(linestyle=':')
plt.title('kNN-Recall versus t')
plt.ylabel('Average recall-KNN')
plt.xlabel('t')
plt.savefig('Q14a2.png',dpi=350,bbox_inches='tight')
plt.show()
```



In [ ]:

```
plt.plot(rec_list_KNN,prec_list_KNN,linestyle='--',color='r')
plt.grid(linestyle=':')
plt.title('kNN- Precision vs. Recall')
plt.ylabel('Average precision-KNN')
plt.xlabel('Average recall-KNN')
plt.savefig('Q14a3.png',dpi=300,bbox_inches='tight')
plt.show()
```



In [ ]:

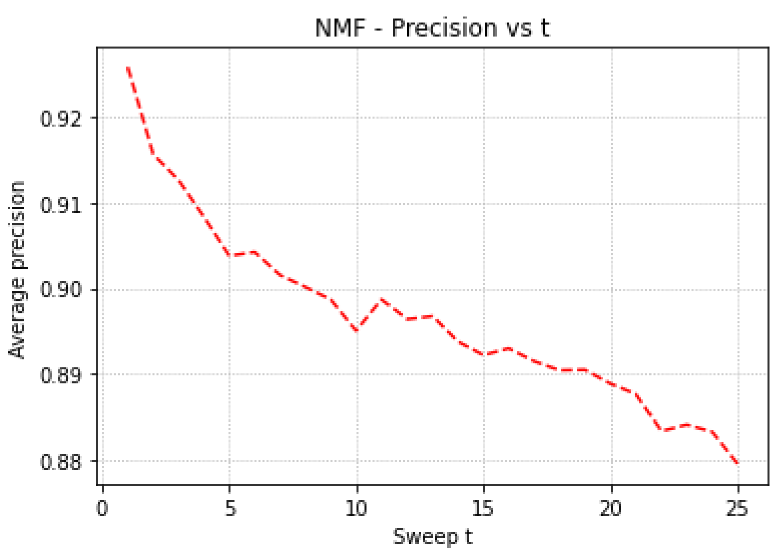
```
t = np.arange(1,26,1)
k_fold = KFold(n_splits=10)

prec_list_NMF = []
rec_list_NMF = []
for val in t:
    print('Iterating for the value of t =',val)
    precision_Local_set = []
    recall_Local_set = []
    for Train_list, Test_list in k_fold.split(Dataset_Ratings):
        G = {}
        for row in Test_list:
            if row[0] in G.keys():
                if row[2] >= 3.0:
                    G[row[0]].add(row[1])
            else:
                G[row[0]] = set()
                if row[2] >= 3.0:
                    G[row[0]].add(row[1])
        dict_of_items = {}
        for row in Test_list:
            if row[0] in dict_of_items.keys():
                dict_of_items[row[0]].append(row[1])
            else:
                dict_of_items[row[0]] = []
                dict_of_items[row[0]].append(row[1])
        NMF_mod_testset = [row for row in Test_list if (len(dict_of_items[row[0]
        res = NMF(n_factors=18,n_epochs=50,verbose=False).fit(Train_list).test(N
        Est_Ratings = {}
        for row in res:
            if row[0] in Est_Ratings.keys():
                Est_Ratings[row[0]].append((row[1],row[3]))
            else:
                Est_Ratings[row[0]] = []
                Est_Ratings[row[0]].append((row[1],row[3]))
        precision_u = []
        recall_u = []
        for item in Est_Ratings.keys():
            Set_all = Est_Ratings[item]
            Set_all = sorted(Set_all,key=lambda x:x[1],reverse=True)
            Set_t = set([row[0] for row in Set_all[0:val]])
```

```
            precision_u.append(len(Set_t.intersection(G[item]))/float(len(Set_t)
            recall_u.append(len(Set_t.intersection(G[item]))/float(len(G[item]))
        precision_Local_set.append(np.mean(precision_u))
        recall_Local_set.append(np.mean(recall_u))
    prec_list_NMF.append(np.mean(precision_Local_set))
    rec_list_NMF.append(np.mean(recall_Local_set))
```

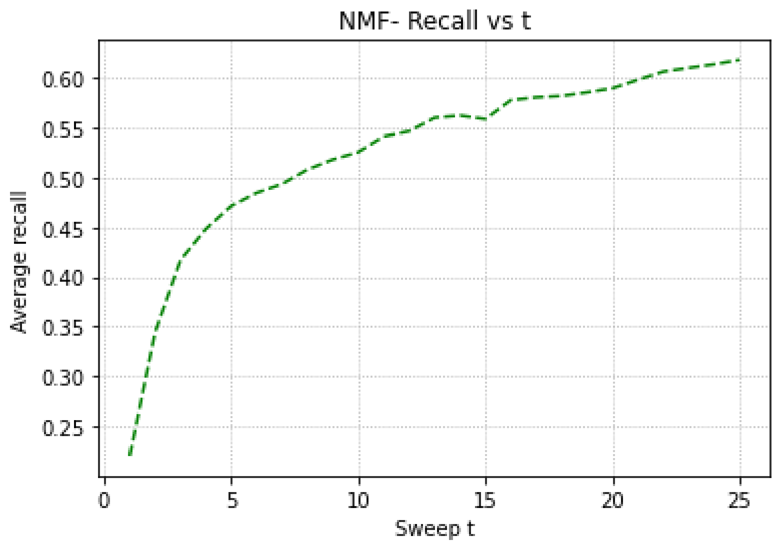
```
Iterating for the value of t = 1
Iterating for the value of t = 2
Iterating for the value of t = 3
Iterating for the value of t = 4
Iterating for the value of t = 5
Iterating for the value of t = 6
Iterating for the value of t = 7
Iterating for the value of t = 8
Iterating for the value of t = 9
Iterating for the value of t = 10
Iterating for the value of t = 11
Iterating for the value of t = 12
Iterating for the value of t = 13
Iterating for the value of t = 14
Iterating for the value of t = 15
Iterating for the value of t = 16
Iterating for the value of t = 17
Iterating for the value of t = 18
Iterating for the value of t = 19
Iterating for the value of t = 20
Iterating for the value of t = 21
Iterating for the value of t = 22
Iterating for the value of t = 23
Iterating for the value of t = 24
Iterating for the value of t = 25
```

In [ ]:

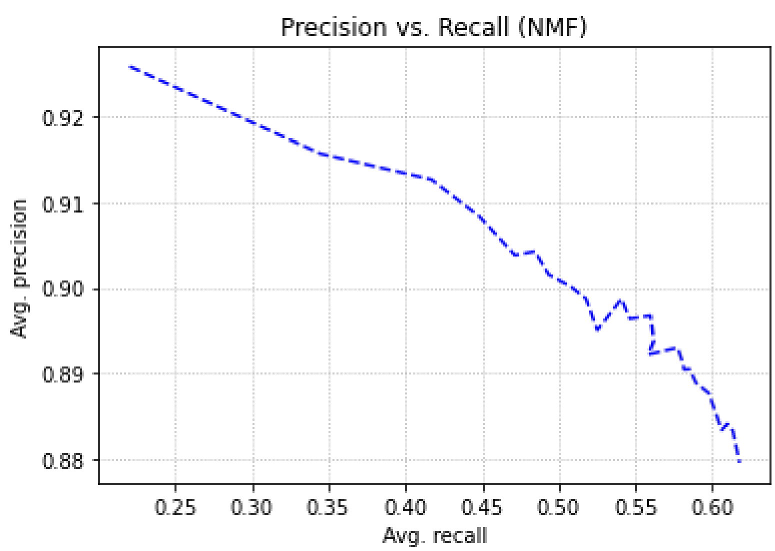
```
plt.plot(t,prec_list_NMF,linestyle='--',color='r')
plt.grid(linestyle=':')
plt.title('NMF - Precision vs t')
plt.ylabel('Average precision')
plt.xlabel('Sweep t')
plt.savefig('Q14b1.png',dpi=350,bbox_inches='tight')
plt.show()
```



In [ ]:
```
plt.plot(t,rec_list_NMF,linestyle='--',color='g')
plt.grid(linestyle=':')
plt.title('NMF- Recall vs t')
plt.ylabel('Average recall')
plt.xlabel('Sweep t')
plt.savefig('Q14b2.png',dpi=350,bbox_inches='tight')
plt.show()
```



In [ ]:
```
plt.plot(rec_list_NMF,prec_list_NMF,linestyle='--',color='b')
plt.grid(linestyle=':')
plt.title('Precision vs. Recall (NMF)')
plt.ylabel('Avg. precision')
plt.xlabel('Avg. recall')
plt.savefig('Q14b3.png',dpi=350,bbox_inches='tight')
plt.show()
```



In [ ]:
```
#High Variance
t = np.arange(1,26,1)
k_fold = KFold(n_splits=10)
```

In [ ]:
```
prec_list_SVD = []
```

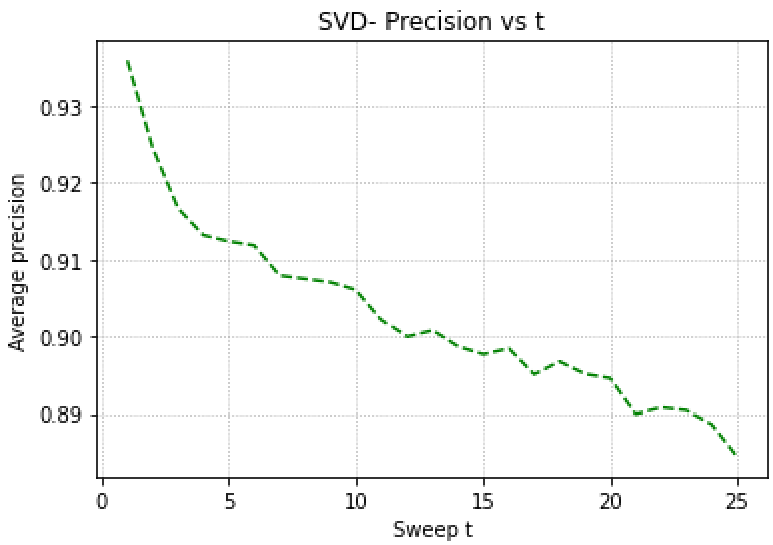
```
rec_list_SVD = []
for val in t:
    print('Iteration for the value of t =',val)
    precision_Local_set = []
    recall_Local_set = []
    for Train_list, Test_list in k_fold.split(Dataset_Ratings):
        D = {}
        for row in Test_list:
            if row[0] in D.keys():
                if row[2] >= 3.0:
                    D[row[0]].add(row[1])
            else:
                D[row[0]] = set()
                if row[2] >= 3.0:
                    D[row[0]].add(row[1])
        dict_of_items = {}
        for row in Test_list:
            if row[0] in dict_of_items.keys():
                dict_of_items[row[0]].append(row[1])
            else:
                dict_of_items[row[0]] = []
                dict_of_items[row[0]].append(row[1])
        SVD_mod_testset = [row for row in Test_list if (len(dict_of_items[row[0]
        res = SVD(n_factors=20,n_epochs=20,verbose=False).fit(Train_list).test(S
        Est_Ratings = {}
        for row in res:
            if row[0] in Est_Ratings.keys():
                Est_Ratings[row[0]].append((row[1],row[3]))
            else:
                Est_Ratings[row[0]] = []
                Est_Ratings[row[0]].append((row[1],row[3]))
        precision_u = []
        recall_u = []
        for item in Est_Ratings.keys():
            Set_all = Est_Ratings[item]
            Set_all = sorted(Set_all,key=lambda x:x[1],reverse=True)
            Set_t = set([row[0] for row in Set_all[0:val]])
            precision_u.append(len(Set_t.intersection(D[item]))/float(len(Set_t)
            recall_u.append(len(Set_t.intersection(D[item]))/float(len(D[item]))
        precision_Local_set.append(np.mean(precision_u))
        recall_Local_set.append(np.mean(recall_u))
    prec_list_SVD.append(np.mean(precision_Local_set))
    rec_list_SVD.append(np.mean(recall_Local_set))
```

```
Iteration for the value of t = 1
Iteration for the value of t = 2
Iteration for the value of t = 3
Iteration for the value of t = 4
Iteration for the value of t = 5
Iteration for the value of t = 6
Iteration for the value of t = 7
Iteration for the value of t = 8
Iteration for the value of t = 9
Iteration for the value of t = 10
Iteration for the value of t = 11
Iteration for the value of t = 12
Iteration for the value of t = 13
Iteration for the value of t = 14
Iteration for the value of t = 15
Iteration for the value of t = 16
```

```
Iteration for the value of t = 17
Iteration for the value of t = 18
Iteration for the value of t = 19
Iteration for the value of t = 20
Iteration for the value of t = 21
Iteration for the value of t = 22
Iteration for the value of t = 23
Iteration for the value of t = 24
Iteration for the value of t = 25
```

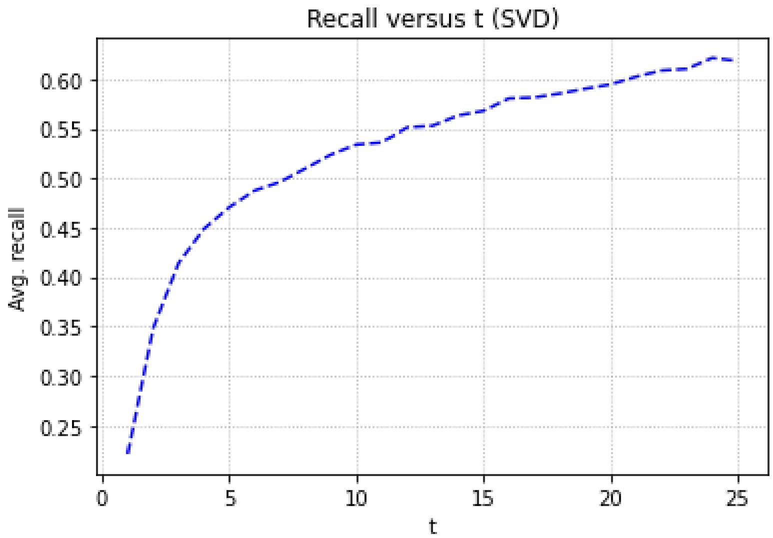
In [ ]:

```
plt.plot(t,prec_list_SVD,linestyle='--',color='g')
plt.grid(linestyle=':')
plt.title('SVD- Precision vs t')
plt.ylabel('Average precision')
plt.xlabel('Sweep t')
plt.savefig('Q14c1.png',dpi=350,bbox_inches='tight')
plt.show()
```



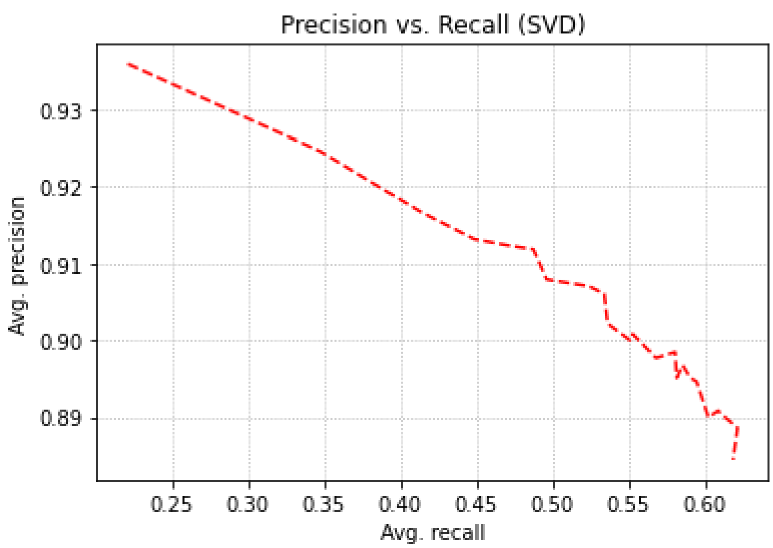
In [ ]:

```
plt.plot(t,rec_list_SVD,linestyle='--',color='b')
plt.grid(linestyle=':')
plt.title('Recall versus t (SVD)')
plt.ylabel('Avg. recall')
plt.xlabel('t')
plt.savefig('Q14c2.png',dpi=350,bbox_inches='tight')
plt.show()
```



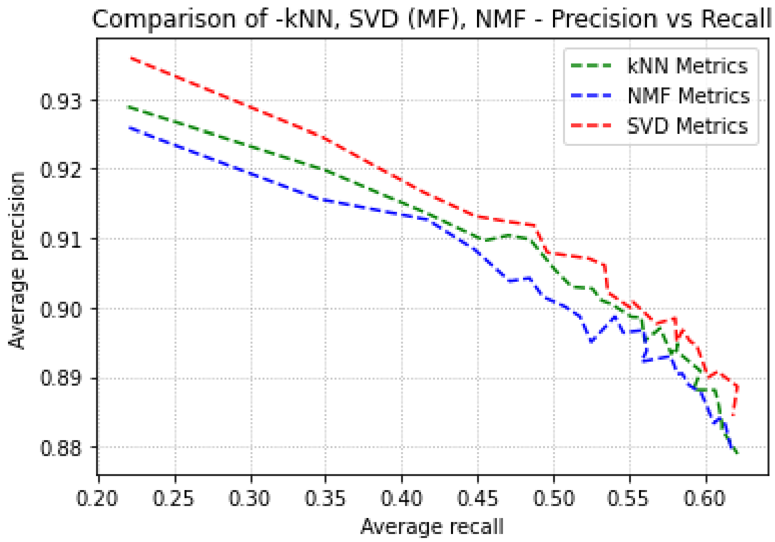
In [ ]:

```
plt.plot(rec_list_SVD,prec_list_SVD,linestyle='--',color='r')
plt.grid(linestyle=':')
plt.title('Precision vs. Recall (SVD)')
plt.ylabel('Avg. precision')
plt.xlabel('Avg. recall')
plt.savefig('Q14c3.png',dpi=300,bbox_inches='tight')
plt.show()
```



In [ ]:

```
fig, ax = plt.subplots()
ax.plot(rec_list_KNN,prec_list_KNN,linestyle='--',color='g',label='kNN Metrics')
ax.plot(rec_list_NMF,prec_list_NMF,linestyle='--',color='b',label='NMF Metrics')
ax.plot(rec_list_SVD,prec_list_SVD,linestyle='--',color='r',label='SVD Metrics')
plt.grid(linestyle=':')
plt.title('Comparison of -kNN, SVD (MF), NMF - Precision vs Recall')
plt.ylabel('Average precision')
plt.xlabel('Average recall')
plt.legend(loc="best")
plt.savefig('Q14d.png',dpi=350,bbox_inches='tight')
plt.show()
```


Comparison of -kNN, SVD (MF), NMF - Precision vs Recall
kNN Metrics
NMF Metrics
SVD Metrics
Average precision
Average recall
0.93
0.92
0.91
0.90
0.89
0.88
0.20
0.25
0.30
0.35
0.40
0.45
0.50
0.55
0.60

In [ ]: