

```
In [1]: import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings("ignore")
```

```
In [2]: treatment=pd.read_csv(r"C:\Users\reshma_koduri\Downloads\treatment ratings.csv")
```

```
In [3]: treatment
```

```
Out[3]:
```

	USER_ID	AGE	SEX	SOURCE	HAEMOGLOBINS	ERYTHROCYTE	LEUCOCYTE	HAEMATOCRIT
0	1	68	F	out	11.8	4.65	6.3	35.1
1	2	70	F	out	14.8	5.39	12.7	43.5
2	3	64	F	out	11.3	4.74	13.2	33.5
3	4	73	F	out	13.7	4.98	10.5	39.1
4	5	66	M	out	9.9	4.23	22.1	30.9
...	...	...	...	...	...	...	...	...
1184	1185	90	F	out	12.5	4.41	10.9	37.1
1185	1186	90	F	out	10.6	4.44	9.4	33.7
1186	1187	90	F	in	11.2	4.06	8.4	35.4
1187	1188	25	M	in	13.8	4.88	3.5	40.0
1188	1189	31	M	in	13.8	4.86	2.3	39.4

1189 rows × 11 columns



```
In [4]: treatment.describe()
```

```
Out[4]:
```

	USER_ID	AGE	HAEMOGLOBINS	ERYTHROCYTE	LEUCOCYTE	HAEMATOCRIT	R
count	1189.000000	1189.000000	1189.000000	1189.000000	1189.000000	1189.000000	1189
mean	595.000000	47.915055	13.158621	4.890294	7.475610	39.529521	2
std	343.379042	20.320298	1.869245	0.672515	3.994819	5.194292	0
min	1.000000	19.000000	3.800000	2.150000	1.100000	14.600000	1
25%	298.000000	31.000000	12.000000	4.480000	4.900000	36.300000	1
50%	595.000000	38.000000	13.200000	4.880000	6.900000	39.500000	2
75%	892.000000	69.000000	14.400000	5.300000	9.200000	43.200000	3
max	1189.000000	96.000000	18.100000	7.610000	32.000000	54.000000	3



In [5]:

```
treatment.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1189 entries, 0 to 1188
Data columns (total 11 columns):
 #   Column                Non-Null Count  Dtype  
---  -
 0   USER_ID              1189 non-null   int64  
 1   AGE                  1189 non-null   int64  
 2   SEX                  1189 non-null   object  
 3   SOURCE               1189 non-null   object  
 4   HAEMOGLOBINS         1189 non-null   float64 
 5   ERYTHROCYTE         1189 non-null   float64 
 6   LEUCOCYTE            1189 non-null   float64 
 7   HAEMATOCRIT          1189 non-null   float64 
 8   TREATMENT            1189 non-null   object  
 9   CONDITION            1189 non-null   object  
10   RATINGS              1189 non-null   int64  
dtypes: float64(4), int64(3), object(4)
memory usage: 102.3+ KB
```

In [6]:

```
treatment.isna().sum()
```

Out[6]:

```
USER_ID      0
AGE          0
SEX          0
SOURCE       0
HAEMOGLOBINS 0
ERYTHROCYTE 0
LEUCOCYTE    0
HAEMATOCRIT  0
TREATMENT    0
CONDITION    0
RATINGS      0
dtype: int64
```

In [7]:

```
treatment.groupby(['CONDITION']).count()
```

Out[7]:

	USER_ID	AGE	SEX	SOURCE	HAEMOGLOBINS	ERYTHROCYTE	LEUCOCYTE	HAEMATOCRIT
CONDITION								
<b>improved</b>	414	414	414	414	414	414	414	
<b>stable</b>	418	418	418	418	418	418	418	
<b>worse</b>	357	357	357	357	357	357	357	



In [8]:

```
treatment['CONDITION'].unique()
```

Out[8]:

```
array(['worse', 'improved', 'stable'], dtype=object)
```

In [9]:

```
treatment['SEX'].unique()
```

Out[9]:

```
array(['F', 'M'], dtype=object)
```

```
In [10]: treatment['SOURCE'].unique()
```

```
Out[10]: array(['out', 'in'], dtype=object)
```

```
In [11]: treatment['TREATMENT'].unique()
```

```
Out[11]: array(['Oxygen Therapy', 'Convalescent Plasma', 'COVID-19 Vaccines',
        'Tocilizumab', 'Dexamethasone',
        'Non-invasive Ventilation (CPAP/BiPAP)',
        'Casirivimab and imdevimab', 'Baricitinib', 'Remdesivir',
        'Monoclonal Antibodies', 'Etesevimab', 'Mechanical Ventilation',
        'Molnupiravir', 'Bamlanivimab', 'Sotrovimab', 'Anticoagulants',
        'Methylprednisolone', 'Heparin'], dtype=object)
```

```
In [12]: treatment['SEX']=treatment['SEX'].map({'F':1,'M':0})
        treatment['CONDITION']=treatment['CONDITION'].map({'worse':1,'stable':2,'improved':3})
        treatment['SOURCE']=treatment['SOURCE'].map({'in':1,'out':0})
```

```
In [13]: treatment
```

```
Out[13]:
```

	USER_ID	AGE	SEX	SOURCE	HAEMOGLOBINS	ERYTHROCYTE	LEUCOCYTE	HAEMATOCRIT
0	1	68	1	0	11.8	4.65	6.3	35.1
1	2	70	1	0	14.8	5.39	12.7	43.5
2	3	64	1	0	11.3	4.74	13.2	33.5
3	4	73	1	0	13.7	4.98	10.5	39.1
4	5	66	0	0	9.9	4.23	22.1	30.9
...	...	...	...	...	...	...	...	...
1184	1185	90	1	0	12.5	4.41	10.9	37.1
1185	1186	90	1	0	10.6	4.44	9.4	33.7
1186	1187	90	1	1	11.2	4.06	8.4	35.4
1187	1188	25	0	1	13.8	4.88	3.5	40.0
1188	1189	31	0	1	13.8	4.86	2.3	39.4

1189 rows × 11 columns



```
In [14]: treatment1=treatment.drop(['TREATMENT'],axis=1)
        treatment1
```

```
Out[14]:
```

	USER_ID	AGE	SEX	SOURCE	HAEMOGLOBINS	ERYTHROCYTE	LEUCOCYTE	HAEMATOCRIT
0	1	68	1	0	11.8	4.65	6.3	35.1
1	2	70	1	0	14.8	5.39	12.7	43.5

	USER_ID	AGE	SEX	SOURCE	HAEMOGLOBINS	ERYTHROCYTE	LEUCOCYTE	HAEMATOCRIT	C
	2	3	64	1	0	11.3	4.74	13.2	33.5
	3	4	73	1	0	13.7	4.98	10.5	39.1
	4	5	66	0	0	9.9	4.23	22.1	30.9
	...	...	...	...	...	...	...	...	...
	1184	1185	90	1	0	12.5	4.41	10.9	37.1
	1185	1186	90	1	0	10.6	4.44	9.4	33.7
	1186	1187	90	1	1	11.2	4.06	8.4	35.4
	1187	1188	25	0	1	13.8	4.88	3.5	40.0
	1188	1189	31	0	1	13.8	4.86	2.3	39.4

1189 rows × 10 columns

```
In [15]: import numpy as np
import seaborn as sns
from matplotlib import pyplot as plt
import warnings
import os
import importlib
warnings.filterwarnings('ignore')
sns.set_theme(color_codes=True)
```

```
In [16]: def check_and_install_library(library_name):
    try:
        importlib.import_module(library_name)
        print(f"{library_name} is already installed.")
    except ImportError:
        print(f"{library_name} is not installed. Installing...")
        try:
            import pip
            pip.main(['install', library_name])
        except:
            print("Error: Failed to install the library. Please install it manually.")
```

```
In [17]: #!pip install scikit-surprise
```

```
In [18]: !pip show scikit-surprise
```

```
Name: scikit-surprise
Version: 1.1.3
Summary: An easy-to-use library for recommender systems.
Home-page: https://surpriselib.com
Author: Nicolas Hug
Author-email: contact@nicolas-hug.com
License: GPLv3+
Location: c:\users\reshma_koduri\anaconda3\lib\site-packages
Requires: joblib, numpy, scipy
Required-by:
```

```
In [19]: import sys
print(sys.path)
import sys
sys.path.append('/path/to/directory')

['C:\\Users\\reshma_koduri', 'C:\\Users\\reshma_koduri\\anaconda3\\python39.zip',
'C:\\Users\\reshma_koduri\\anaconda3\\DLLs', 'C:\\Users\\reshma_koduri\\anaconda3\\li
b', 'C:\\Users\\reshma_koduri\\anaconda3', '', 'C:\\Users\\reshma_koduri\\anaconda3
\\lib\\site-packages', 'C:\\Users\\reshma_koduri\\anaconda3\\lib\\site-packages\\lock
et-0.2.1-py3.9.egg', 'C:\\Users\\reshma_koduri\\anaconda3\\lib\\site-packages\\win3
2', 'C:\\Users\\reshma_koduri\\anaconda3\\lib\\site-packages\\win32\\lib', 'C:\\Users
\\reshma_koduri\\anaconda3\\lib\\site-packages\\Pythonwin', 'C:\\Users\\reshma_koduri
\\anaconda3\\lib\\site-packages\\IPython\\extensions', 'C:\\Users\\reshma_koduri\\.ip
ython']

In [20]: from surprise import Dataset, Reader
from surprise import KNNBasic, accuracy, SVD
from surprise.model_selection import train_test_split

In [21]: #df = pd.DataFrame(treatment)

In [22]: # Assuming you have a DataFrame named 'data' with columns 'User', 'Treatment', and '
reader = Reader(rating_scale=(1,3)) # Define the rating scale
surprise_data = Dataset.load_from_df(treatment[['USER_ID', 'TREATMENT', 'RATINGS']],

In [23]: trainset, testset = train_test_split(surprise_data, test_size=0.3, random_state=42)

In [24]: model = SVD()
model.fit(trainset)

Out[24]: <surprise.prediction_algorithms.matrix_factorization.SVD at 0x1bc0b61b520>

In [25]: predictions = model.test(testset)

In [26]: from surprise.accuracy import rmse

In [27]: accuracy = rmse(predictions)
print(f'RMSE: {accuracy}')
```

RMSE: 0.1020  
RMSE: 0.1019821695834463