Program Code:

```python
# CSC_LAB_04.py > ...
1    # program to read from a file
2    # strName = "c:\\temp\\ratings.txt"
3    strName = "ratings.txt"
4
5    try:
6        file = open(strName, "r")
7        field1 = ""; field2 = ""; field3 = ""
8        field4 = ""; field5 = ""; field6 = ""; field7 = ""
9
10       # Initialize variables
11       total_rating = 0
12       count = 0
13       male_voters = 0
14       female_voters = 0
15       age_40_49_voters = 0
16       recent_ratings_count = 0
17
18       # Lists that will be used to store ages
19       ages = []
20       male_ages = []
21       female_ages = []
22
23       # Skip the header row
24       next(file)
25
26       for line in file:
27
28           fields = line.rstrip().split(",")
29           field1 = fields[0].strip()   # Member ID
30           field2 = float(fields[1].strip())   # Rating
31           field3 = fields[2].strip()   # Crew involvement
32           field4 = fields[3].strip()   # Date rated
33           field5 = fields[4].strip()   # Gender
34           field6 = int(fields[5].strip())   # Age
35           field7 = fields[6].strip()   # Demographics
36
37           # Calculate the total rating and count voters
38           total_rating += field2
39           count += 1
40
41           # Count male/female voters and collect ages
42           ages.append(field6)
43           if field5 == "M":
44               male_voters += 1
45               male_ages.append(field6)
46           elif field5 == "F":
47               female_voters += 1
48               female_ages.append(field6)
49
50           # Count voters in the 40-49 range
```

This first page of code creates the main variables needed and begins to run through the data in the CSV text file. We can also see the first calculations of number of voters beginning on line 42

```python
CSC_LAB_04.py > ...
 51            if 40 <= field6 <= 49:
 52                age_40_49_voters += 1
 53
 54            # Check if the rating was posted in a recent date
 55            month, day = map(int, field4.split("/"))  # Assuming MM/DD format
 56            if month >= 1 and month <= 2:  # Considering January and February as recent
 57                recent_ratings_count += 1
 58
 59            # Print the current line's fields
 60            print(field1 + "\t" + str(field2) + "\t" + field3 + "\t" +
 61                  field4 + "\t" + field5 + "\t" + str(field6) + "\t" + field7)
 62
 63        # Calculate the average rating, rounded to one decimal place
 64        average_rating = round(total_rating / count, 1)
 65
 66        # Get the age ranges by gender and overall
 67        overall_age_range = max(ages, default=0) - min(ages, default=0)
 68        male_age_range = max(male_ages, default=0) - min(male_ages, default=0)
 69        female_age_range = max(female_ages, default=0) - min(female_ages, default=0)
 70
 71        # Calculate the ratio of recent ratings to total ratings
 72        recent_ratings_ratio = recent_ratings_count / count
 73
 74        # Calculate the ratios
 75        male_age_ratio = male_age_range / max(male_voters, 1)
 76        female_age_ratio = female_age_range / max(female_voters, 1)
 77
 78
 79        # Output the results
 80        print()
 81        print("Average movie rating: ", average_rating)
 82        print("Number of male voters: ", male_voters)
 83        print("Number of female voters: ", female_voters)
 84        print("Number of voters aged 40-49: ", age_40_49_voters)
 85        print("\n-------------------------------------------------\n")
 86        print("Overall Age Range: ", overall_age_range)
 87        print(f"Male Age Range Ratio: {male_age_ratio:.2f}")
 88        print(f"Female Age Range Ratio: {female_age_ratio:.2f}")
 89        print("\n-------------------------------------------------\n")
 90
```

The next page of code finishes all our calculations – estimation of when the ratings happened, calculating the average ratings, age ranges, and ratings ratios. It also prints the data table on line 60. The beginning of the final printing of the findings begins on line 80.

The final blocks of code below are the finishing touches on the findings. It checks the relevant ratios and prints the result based on the criteria we created.

```
 91
 92        # AI: can we predict when the movie was released
 93        if recent_ratings_ratio >= 0.75:
 94            print("It seems the movie was most likely released within the last 2 years") # If the ratio is greater than or equal
 95                                                                    # to .75 we assume the movie is newer
 96        elif 0.50 < recent_ratings_ratio < 0.75:
 97            print("SIt seems the movie was most likely released within the last 10 years") #If the ratio is between .5 and .75
 98                                                                    # we assume the movie is within the decade
 99        else:
100            print("It seems the movie was most likely released within the last 15 years or older") # If the ratio is less than
101                                                                    # .5 we assume the movie is within the last 15 years or more
102
103        print("\n-------------------------------------------------\n")
104
105
106        # AI: can we predict the genre?
107        if male_age_ratio > female_age_ratio:
108            print("It seems the genre of the movie being rated is Action or Adventure\n") # Assuming men are dominating the votes
109        else:
110            print("It seems the genre of the movie being rated is Romance\n") # Assuming woman are dominating the votes
111
112        print("\n-------------------------------------------------\n")
113
114        file.close()
115
116 except IOError:
117     print("File appears to not exist!")
118
```

Program Output:

```
PS C:\Users\dante\Intro to AI\LAB_4>  c:; cd 'c:\Users\dante\Intro to AI\LAB_4'; & 'c:\Users\dante\AppDa
ugpy\launcher' '55071' '--' 'c:\Users\dante\Intro to AI\LAB_4\test.py'
101    5.0    No     01/10  M     40     USA
102    4.0    No     01/10  F     43     USA
106    3.5    Yes    01/11  F     18     Global
100    4.0    No     01/12  M     23     USA
111    5.0    Yes    01/12  M     25     USA
118    2.0    No     01/13  M     19     Global
131    3.5    Yes    01/13  F     49     USA
103    5.0    No     01/13  M     72     USA
107    4.0    No     01/13  F     19     USA
105    1.0    Yes    01/14  F     37     USA
132    5.0    Yes    02/04  M     21     USA
137    3.6    No     02/05  M     29     Global
122    4.5    Yes    02/05  M     23     USA
109    5.0    No     02/06  F     19     USA
112    4.1    Yes    02/06  M     36     USA

Average movie rating:  3.9
Number of male voters:  9
Number of female voters:  6
Number of voters aged 40-49:  3


-------------------------------------------------


Overall Age Range:  54
Male Age Range Ratio: 5.89
Female Age Range Ratio: 5.17


-------------------------------------------------

It seems the movie was most likely released within the last 2 years

-------------------------------------------------


It seems the genre of the movie being rated is Action or Adventure


-------------------------------------------------
```

Artificial Intelligence Reflection:

In this application, we were able to demonstrate concepts of artificial intelligence in unsupervised learning of the CSV data frame given about movie ratings. We were not sure what the movies were, when they were released, or the genre of movie. We used concepts of unsupervised learning to make an educated estimate on what they could be, based on the criteria we created for the machine learning code.