mental-health-x-music

June 18, 2023

# 1 Correlation of Music to Mental Health

This notebook explores the intriguing relationship between music and mental health. This notebook aims to analyze and understand how music can impact our emotional well-being and potentially serve as a therapeutic tool for managing mental health conditions.

[1]:
```python
import pandas as pd

data = pd.read_csv('./data/mxmh_survey_results.csv')
data
```

[1]:
```
               Timestamp   Age Primary streaming service  Hours per day  \
0     8/27/2022 19:29:02  18.0                   Spotify            3.0
1     8/27/2022 19:57:31  63.0                   Pandora            1.5
2     8/27/2022 21:28:18  18.0                   Spotify            4.0
3     8/27/2022 21:40:40  61.0             YouTube Music            2.5
4     8/27/2022 21:54:47  18.0                   Spotify            4.0
..                   …     …                         …            …
731  10/30/2022 14:37:28  17.0                   Spotify            2.0
732   11/1/2022 22:26:42  18.0                   Spotify            1.0
733   11/3/2022 23:24:38  19.0   Other streaming service            6.0
734   11/4/2022 17:31:47  19.0                   Spotify            5.0
735    11/9/2022 1:55:20  29.0             YouTube Music            2.0

    While working Instrumentalist Composer         Fav genre Exploratory  \
0             Yes             Yes      Yes             Latin         Yes
1             Yes              No       No              Rock         Yes
2              No              No       No  Video game music          No
3             Yes              No      Yes              Jazz         Yes
4             Yes              No       No               R&B         Yes
..              …               …        …                 …           …
731           Yes             Yes       No              Rock         Yes
732           Yes             Yes       No               Pop         Yes
733           Yes              No      Yes               Rap         Yes
734           Yes             Yes       No         Classical          No
735           Yes              No       No           Hip hop         Yes

    Foreign languages  …  Frequency [R&B]  Frequency [Rap] Frequency [Rock]  \
```

```
0                 Yes  …        Sometimes  Very frequently            Never
1                  No  …        Sometimes          Rarely  Very frequently
2                 Yes  …            Never          Rarely           Rarely
3                 Yes  …        Sometimes           Never            Never
4                  No  …  Very frequently  Very frequently            Never
..                ...  ...              ...             ...              ...
731               Yes  …            Never          Rarely  Very frequently
732               Yes  …            Never           Never        Sometimes
733                No  …        Sometimes       Sometimes           Rarely
734                No  …            Never           Never            Never
735               Yes  …  Very frequently  Very frequently  Very frequently

    Frequency [Video game music] Anxiety Depression Insomnia  OCD  \
0                      Sometimes     3.0        0.0      1.0  0.0
1                         Rarely     7.0        2.0      2.0  1.0
2                Very frequently     7.0        7.0     10.0  2.0
3                          Never     9.0        7.0      3.0  3.0
4                         Rarely     7.0        2.0      5.0  9.0
..                           ...     ...        ...      ...  ...
731                        Never     7.0        6.0      0.0  9.0
732                    Sometimes     3.0        2.0      2.0  5.0
733                       Rarely     2.0        2.0      2.0  2.0
734                    Sometimes     2.0        3.0      2.0  1.0
735                       Rarely     2.0        2.0      2.0  5.0

    Music effects    Permissions
0             NaN  I understand.
1             NaN  I understand.
2       No effect  I understand.
3         Improve  I understand.
4         Improve  I understand.
..            ...            ...
731       Improve  I understand.
732       Improve  I understand.
733       Improve  I understand.
734       Improve  I understand.
735       Improve  I understand.

[736 rows x 33 columns]
```

Now, show the data table head.

[2]:
```
data.head()
```

[2]:
```
             Timestamp   Age Primary streaming service  Hours per day  \
0  8/27/2022 19:29:02  18.0                   Spotify            3.0
1  8/27/2022 19:57:31  63.0                   Pandora            1.5
2  8/27/2022 21:28:18  18.0                   Spotify            4.0
```

```
3  8/27/2022 21:40:40  61.0            YouTube Music            2.5
4  8/27/2022 21:54:47  18.0                  Spotify            4.0

  While working Instrumentalist Composer         Fav genre Exploratory  \
0           Yes              Yes      Yes             Latin         Yes
1           Yes               No       No              Rock         Yes
2            No               No       No  Video game music          No
3           Yes               No      Yes              Jazz         Yes
4           Yes               No       No               R&B         Yes

  Foreign languages  …  Frequency [R&B]  Frequency [Rap] Frequency [Rock]  \
0               Yes  …        Sometimes  Very frequently            Never
1                No  …        Sometimes           Rarely  Very frequently
2               Yes  …            Never           Rarely           Rarely
3               Yes  …        Sometimes            Never            Never
4                No  …  Very frequently  Very frequently            Never

  Frequency [Video game music] Anxiety Depression Insomnia  OCD Music effects  \
0                    Sometimes     3.0        0.0      1.0  0.0           NaN
1                       Rarely     7.0        2.0      2.0  1.0           NaN
2              Very frequently     7.0        7.0     10.0  2.0     No effect
3                        Never     9.0        7.0      3.0  3.0       Improve
4                       Rarely     7.0        2.0      5.0  9.0       Improve

     Permissions
0  I understand.
1  I understand.
2  I understand.
3  I understand.
4  I understand.

[5 rows x 33 columns]
```

## 1.1 Data Shape

Showing the data's shape in this study correlating the mental health and music is essential for understanding data distribution, identifying outliers, assessing assumptions, communicating findings, and supporting the interpretation of results. It enhances the validity and reliability of the study and enables researchers to draw meaningful conclusions from the data analysis.

[3]:
```
data.shape
```

[3]:
```
(736, 33)
```

## 1.2 Data Overview

Using the `data.info()` method in your study on the correlation between mental health and music can provide valuable information about the dataset.

[4]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 736 entries, 0 to 735
Data columns (total 33 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   Timestamp                     736 non-null    object
 1   Age                           735 non-null    float64
 2   Primary streaming service     735 non-null    object
 3   Hours per day                 736 non-null    float64
 4   While working                 733 non-null    object
 5   Instrumentalist               732 non-null    object
 6   Composer                      735 non-null    object
 7   Fav genre                     736 non-null    object
 8   Exploratory                   736 non-null    object
 9   Foreign languages             732 non-null    object
 10  BPM                           629 non-null    float64
 11  Frequency [Classical]         736 non-null    object
 12  Frequency [Country]           736 non-null    object
 13  Frequency [EDM]               736 non-null    object
 14  Frequency [Folk]              736 non-null    object
 15  Frequency [Gospel]            736 non-null    object
 16  Frequency [Hip hop]           736 non-null    object
 17  Frequency [Jazz]              736 non-null    object
 18  Frequency [K pop]             736 non-null    object
 19  Frequency [Latin]             736 non-null    object
 20  Frequency [Lofi]              736 non-null    object
 21  Frequency [Metal]             736 non-null    object
 22  Frequency [Pop]               736 non-null    object
 23  Frequency [R&B]               736 non-null    object
 24  Frequency [Rap]               736 non-null    object
 25  Frequency [Rock]              736 non-null    object
 26  Frequency [Video game music]  736 non-null    object
 27  Anxiety                       736 non-null    float64
 28  Depression                    736 non-null    float64
 29  Insomnia                      736 non-null    float64
 30  OCD                           736 non-null    float64
 31  Music effects                 728 non-null    object
 32  Permissions                   736 non-null    object
dtypes: float64(7), object(26)
memory usage: 189.9+ KB
```

[5]:
```
data.dtypes
```

[5]:
```
Timestamp                        object
Age                             float64
Primary streaming service        object
```

```
Hours per day                   float64
While working                    object
Instrumentalist                  object
Composer                         object
Fav genre                        object
Exploratory                      object
Foreign languages                object
BPM                             float64
Frequency [Classical]            object
Frequency [Country]              object
Frequency [EDM]                  object
Frequency [Folk]                 object
Frequency [Gospel]               object
Frequency [Hip hop]              object
Frequency [Jazz]                 object
Frequency [K pop]                object
Frequency [Latin]                object
Frequency [Lofi]                 object
Frequency [Metal]                object
Frequency [Pop]                  object
Frequency [R&B]                  object
Frequency [Rap]                  object
Frequency [Rock]                 object
Frequency [Video game music]     object
Anxiety                         float64
Depression                      float64
Insomnia                        float64
OCD                             float64
Music effects                    object
Permissions                      object
dtype: object
```

[6]:
```python
data.describe().T
```

[6]:
```
               count          mean           std   min    25%    50%    75%  \
Age            735.0  2.520680e+01  1.205497e+01  10.0   18.0   21.0   28.0
Hours per day  736.0  3.572758e+00  3.028199e+00   0.0    2.0    3.0    5.0
BPM            629.0  1.589948e+06  3.987261e+07   0.0  100.0  120.0  144.0
Anxiety        736.0  5.837636e+00  2.793054e+00   0.0    4.0    6.0    8.0
Depression     736.0  4.796196e+00  3.028870e+00   0.0    2.0    5.0    7.0
Insomnia       736.0  3.738451e+00  3.088689e+00   0.0    1.0    3.0    6.0
OCD            736.0  2.637228e+00  2.842017e+00   0.0    0.0    2.0    5.0

                       max
Age                   89.0
Hours per day         24.0
BPM            999999999.0
```

```
Anxiety              10.0
Depression           10.0
Insomnia             10.0
OCD                  10.0
```

As shown in the table above, the maximum hours per day listened to music is 24 hours (a whole day). Meanwhile, the youngest from the data is 10, and the oldest is 89.

[7]:
```python
data.describe(exclude='number').T
```

[7]:

| | count | unique | top | freq |
|---|---|---|---|---|
| Timestamp | 736 | 735 | 8/28/2022 16:15:08 | 2 |
| Primary streaming service | 735 | 6 | Spotify | 458 |
| While working | 733 | 2 | Yes | 579 |
| Instrumentalist | 732 | 2 | No | 497 |
| Composer | 735 | 2 | No | 609 |
| Fav genre | 736 | 16 | Rock | 188 |
| Exploratory | 736 | 2 | Yes | 525 |
| Foreign languages | 732 | 2 | Yes | 404 |
| Frequency [Classical] | 736 | 4 | Rarely | 259 |
| Frequency [Country] | 736 | 4 | Never | 343 |
| Frequency [EDM] | 736 | 4 | Never | 307 |
| Frequency [Folk] | 736 | 4 | Never | 292 |
| Frequency [Gospel] | 736 | 4 | Never | 535 |
| Frequency [Hip hop] | 736 | 4 | Sometimes | 218 |
| Frequency [Jazz] | 736 | 4 | Never | 261 |
| Frequency [K pop] | 736 | 4 | Never | 416 |
| Frequency [Latin] | 736 | 4 | Never | 443 |
| Frequency [Lofi] | 736 | 4 | Never | 280 |
| Frequency [Metal] | 736 | 4 | Never | 264 |
| Frequency [Pop] | 736 | 4 | Very frequently | 277 |
| Frequency [R&B] | 736 | 4 | Never | 225 |
| Frequency [Rap] | 736 | 4 | Rarely | 215 |
| Frequency [Rock] | 736 | 4 | Very frequently | 330 |
| Frequency [Video game music] | 736 | 4 | Never | 236 |
| Music effects | 728 | 3 | Improve | 542 |
| Permissions | 736 | 1 | I understand. | 736 |

The timestamp has the most unique counts because it contains different strings.

### 1.3 Cleaning up the data

[8]:
```python
data = data.drop(data[(data['Age'] > 60)].index, axis=0)
data = data.drop(data[data['Hours per day'] >= 15].index, axis=0)
```

The code above will drop all the listeners who are above 60 and the listeners who listens to music more than or equal 15 hours.

[9]:
```python
data.drop(['Timestamp', 'Permissions'], axis=1, inplace=True)
```

We have now omitted both the column timestamp and permission, because both are irrelevant. Now, let's remove all the missing value.

[10]:
```
data.isnull().sum()

data['Age'] = data['Age'].fillna(round(data['Age'].mean(), 0))
data['Primary streaming service'] = data['Primary streaming service'].
↪fillna(data['Primary streaming service'].mode()[0])
data['While working'] = data['While working'].fillna(data['While working'].
↪mode()[0])
data['Instrumentalist'] = data['Instrumentalist'].
↪fillna(data['Instrumentalist'].mode()[0])
data['Composer'] = data['Composer'].fillna(data['Composer'].mode()[0])
data['Foreign languages'] = data['Foreign languages'].fillna(data['Foreign␣
↪languages'].mode()[0])
data['Music effects'] = data['Music effects'].fillna(data['Music effects'].
↪mode()[0])
```

After cleaning up the data, we can now check the correlation.

[11]:
```
correlation = data.corr()['Age']
correlation.sort_values()
```

[11]:
```
Anxiety         -0.161227
Hours per day   -0.109664
OCD             -0.092071
Depression      -0.078018
BPM             -0.033482
Insomnia         0.039889
Age              1.000000
Name: Age, dtype: float64
```

Let's print all the rows with a missing BPM column.

[12]:
```
data[data['BPM'].isnull() == True]
```

[12]:
```
      Age          Primary streaming service  Hours per day While working  \
10    18.0                           Spotify            3.0           Yes
12    24.0                           Spotify            3.0           Yes
15    17.0                           Spotify            2.0            No
30    20.0                       Apple Music            5.0           Yes
32    19.0                           Spotify            6.0           Yes
..     …                                 …            …           …
688   18.0                           Spotify            4.0           Yes
700   20.0                     YouTube Music            1.0           Yes
706   23.0                           Spotify            1.0           Yes
712   23.0  I do not use a streaming service.            3.0           Yes
717   23.0                           Spotify            2.0            No
```

```
   Instrumentalist Composer Fav genre Exploratory Foreign languages  BPM  \
10             Yes       No   Country         Yes                No  NaN
12              No       No   Hip hop         Yes               Yes  NaN
15              No       No       Pop         Yes               Yes  NaN
30             Yes       No      Rock         Yes               Yes  NaN
32             Yes       No     Metal         Yes               Yes  NaN
..             …        …        …           …                 …   …
688             No       No       R&B          No                No  NaN
700             No       No       Pop          No               Yes  NaN
706            Yes       No      Rock         Yes               Yes  NaN
712             No       No      Rock          No                No  NaN
717             No       No      Rock         Yes               Yes  NaN

    …  Frequency [Pop]  Frequency [R&B] Frequency [Rap] Frequency [Rock]  \
10  …           Rarely           Rarely           Never           Rarely
12  …        Sometimes        Sometimes          Rarely           Rarely
15  …  Very frequently           Rarely       Sometimes        Sometimes
30  …        Sometimes        Sometimes       Sometimes  Very frequently
32  …        Sometimes            Never           Never        Sometimes
..  …              …                …               …                …
688 …        Sometimes  Very frequently       Sometimes            Never
700 …  Very frequently           Rarely       Sometimes           Rarely
706 …  Very frequently        Sometimes       Sometimes  Very frequently
712 …        Sometimes           Rarely           Never  Very frequently
717 …        Sometimes        Sometimes       Sometimes  Very frequently

   Frequency [Video game music] Anxiety Depression Insomnia  OCD  \
10                        Never     7.0        7.0      4.0  7.0
12                        Never     9.0        3.0      2.0  7.0
15                       Rarely     7.0        5.0      4.0  1.0
30                       Rarely     7.0        7.0      2.0  0.0
32                    Sometimes     9.0        8.0      2.0  3.0
..                          …       …          …        …   …
688                       Never     8.0        0.0      0.0  2.0
700             Very frequently     8.0        9.0      6.0  5.0
706             Very frequently     8.0        6.0      1.0  4.0
712                       Never    10.0        5.0      2.0  0.0
717                       Never     5.0        7.0     10.0  2.0

   Music effects
10     No effect
12       Improve
15        Worsen
30       Improve
32       Improve
..           …
```

```
688     No effect
700        Worsen
706       Improve
712       Improve
717     No effect

[101 rows x 31 columns]
```

[13]:
```
sorted(data['Fav genre'].unique())
```

[13]:
```
['Classical',
 'Country',
 'EDM',
 'Folk',
 'Gospel',
 'Hip hop',
 'Jazz',
 'K pop',
 'Latin',
 'Lofi',
 'Metal',
 'Pop',
 'R&B',
 'Rap',
 'Rock',
 'Video game music']
```

The above array is the list of unique favorite genre of the listeners from our data.

[14]:
```
for i in ['Classical', 'Country', 'EDM', 'Folk', 'Gospel', 'Hip hop', 'Jazz',␣
 ↪'K pop', 'Latin', 'Lofi', 'Metal', 'Pop', 'R&B', 'Rap', 'Rock', 'Video game␣
 ↪music']:
    data['BPM'] = data['BPM'].fillna(round(data[data['Fav genre']== i ]['BPM'].
 ↪mean(), 0))
```

### 1.4 Data Visualizations

This section focuses on the visual exploration and representation of data related to mental health and music. Through various plotting techniques and visualizations, this section aims to enhance the understanding of the correlation between mental health and musical factors. Furthermore, this also delves into the creation of more sophisticated visualizations specifically tailored to analyzing the mental health correlation to music.

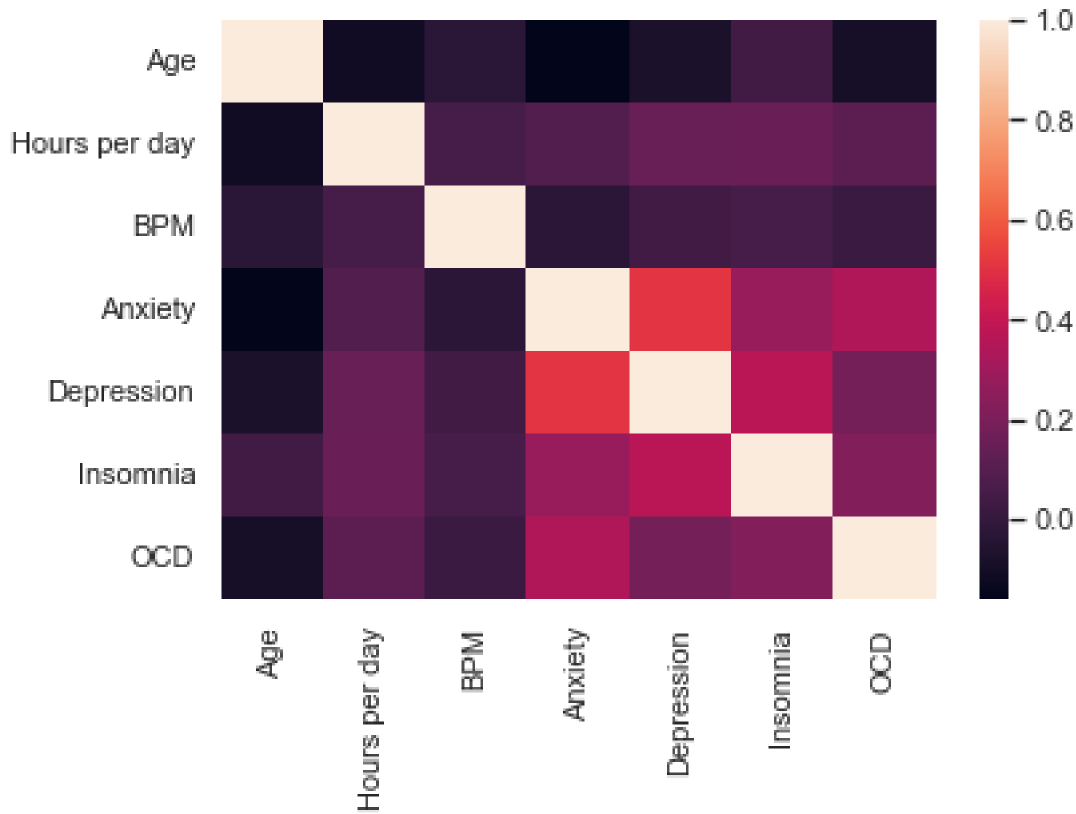
[15]:
```
import seaborn

seaborn.set(color_codes=True)
```

The interpretation of the seaborn `heatmap()` graph in the context of the study on the correlation between mental health and music depends on the specific data being visualized. Generally, a `heatmap()` graph represents the strength or magnitude of the relationship between two variables using a color-coded grid.

[16]:
```
seaborn.heatmap(data.corr())
```

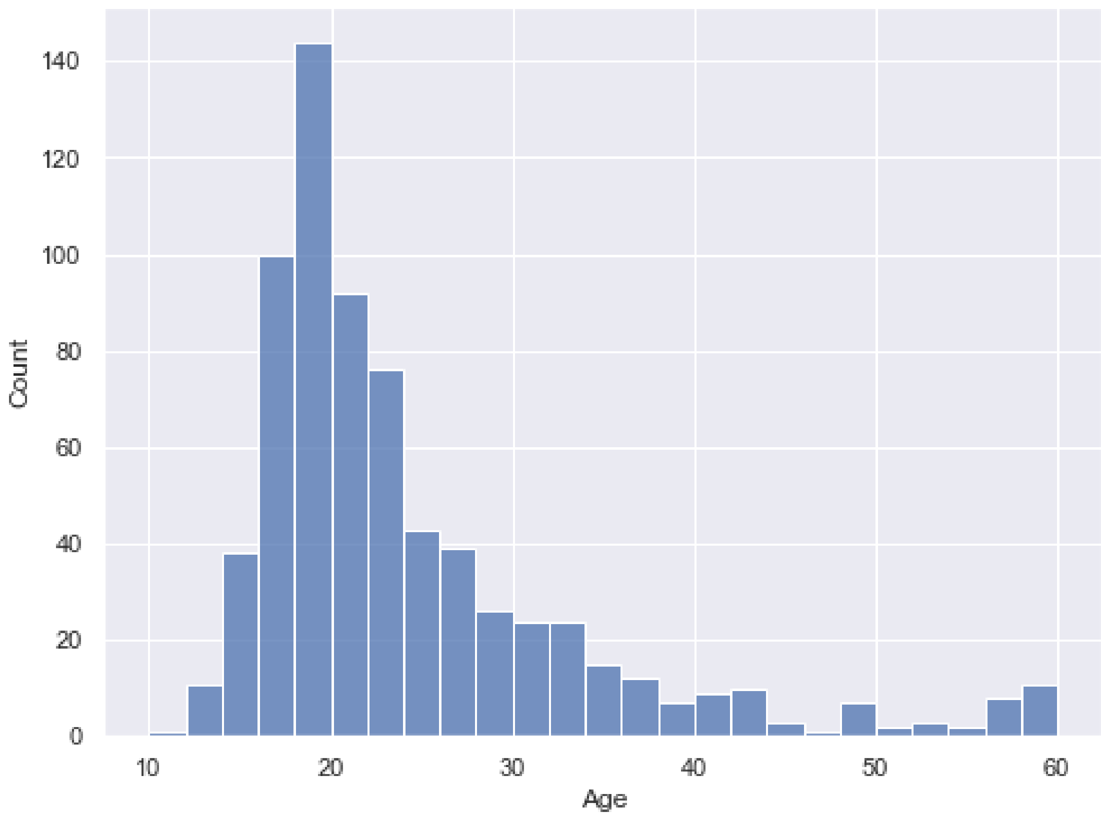
[16]: <AxesSubplot:>




[17]:
```
import matplotlib.pyplot as plt
```

[18]:
```
plt.figure(figsize=(8,6))
seaborn.histplot(data['Age'])
```

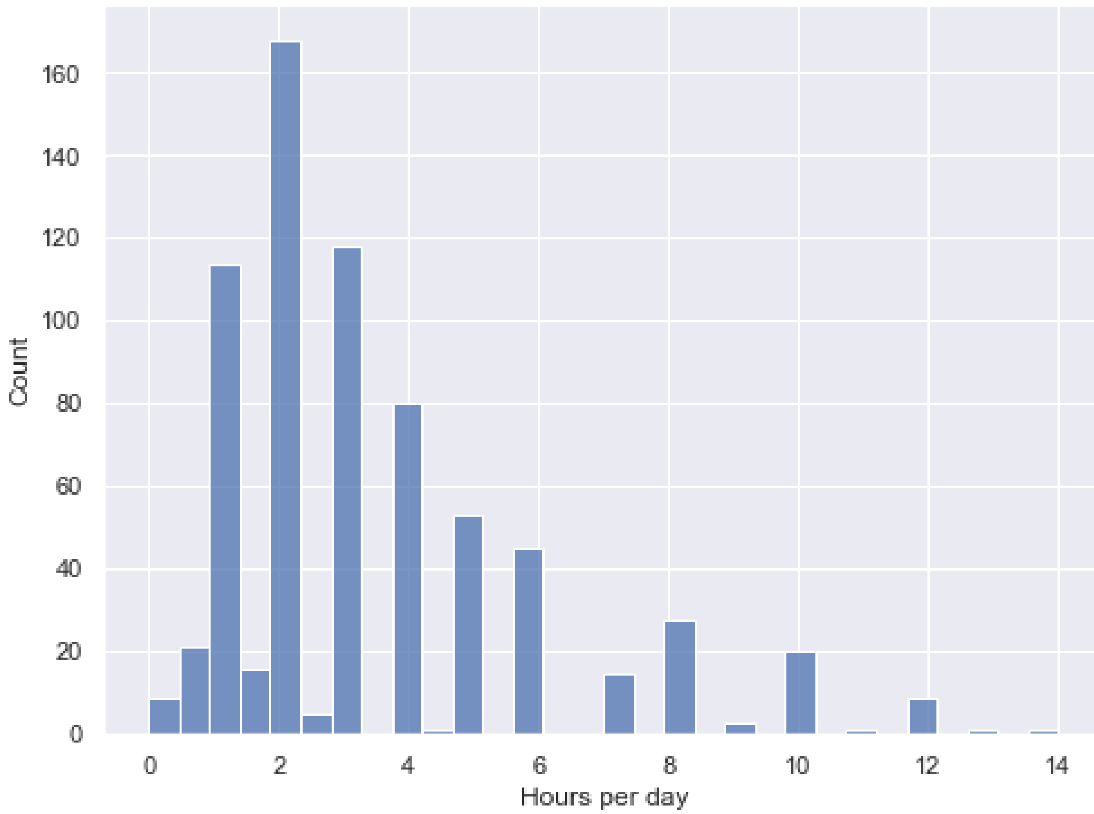
[18]: <AxesSubplot:xlabel='Age', ylabel='Count'>




In average, people from 15 to 24 are much more frequently listening to music.

[19]:
```
plt.figure(figsize=(8,6))
seaborn.histplot(data['Hours per day'])
```

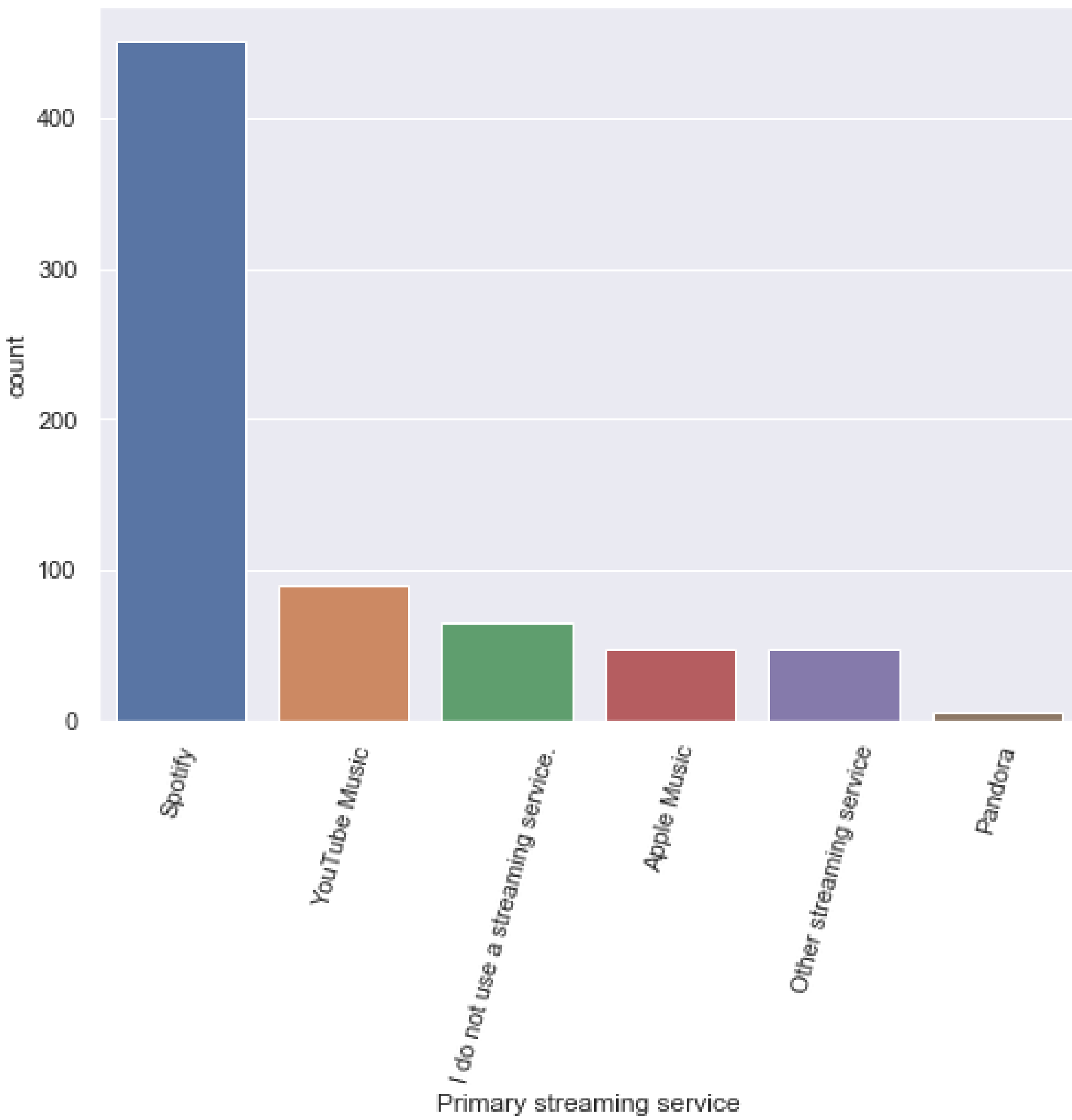
[19]: <AxesSubplot:xlabel='Hours per day', ylabel='Count'>




The people who are working more than 4 hours tend to listen less to music in comparison to those who work less than 4 hours.

[20]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Primary streaming service'])
plt.xticks(rotation=75)
```

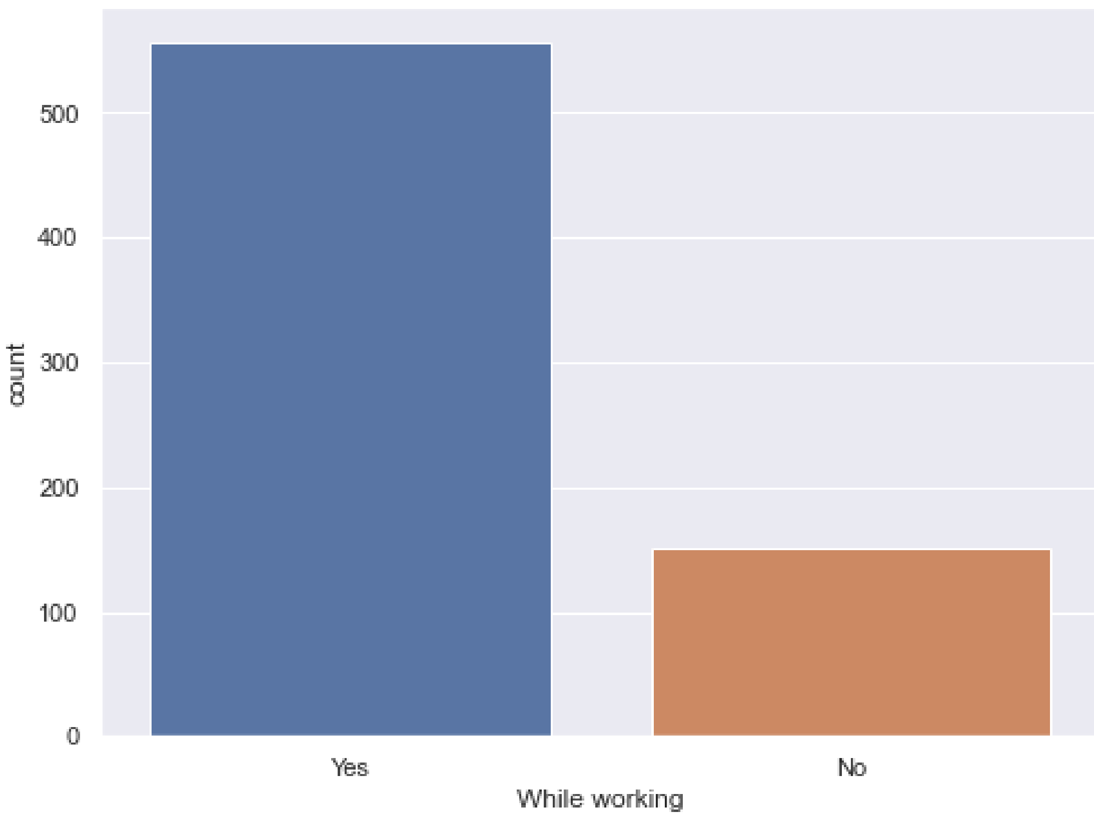
[20]:
```
(array([0, 1, 2, 3, 4, 5]),
 [Text(0, 0, 'Spotify'),
  Text(1, 0, 'YouTube Music'),
  Text(2, 0, 'I do not use a streaming service.'),
  Text(3, 0, 'Apple Music'),
  Text(4, 0, 'Other streaming service'),
  Text(5, 0, 'Pandora')])
```




As shown above, the least used platform is Pandora. While the most popular is the Spotify.

[21]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['While working'])
```

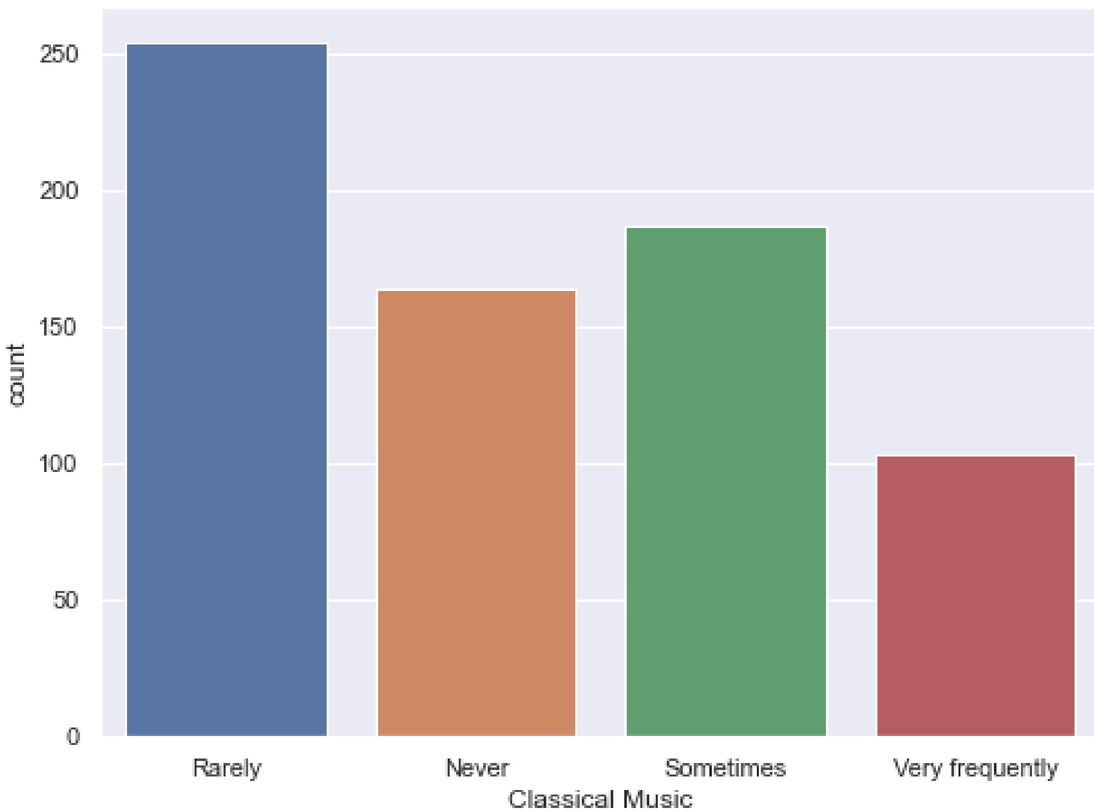
[21]: <AxesSubplot:xlabel='While working', ylabel='count'>




Almost 1/4 of the people doesn't like to listen to music while working.

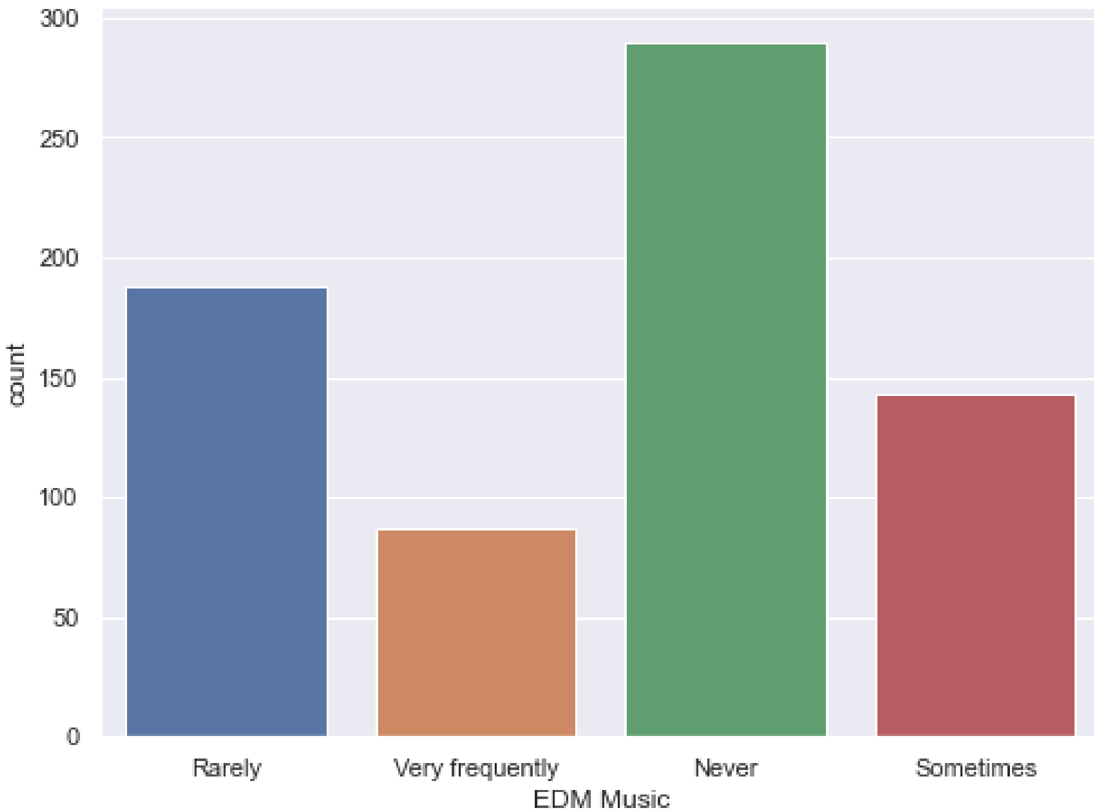
[22]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Classical]'])
plt.xlabel('Classical Music')
```

[22]: Text(0.5, 0, 'Classical Music')



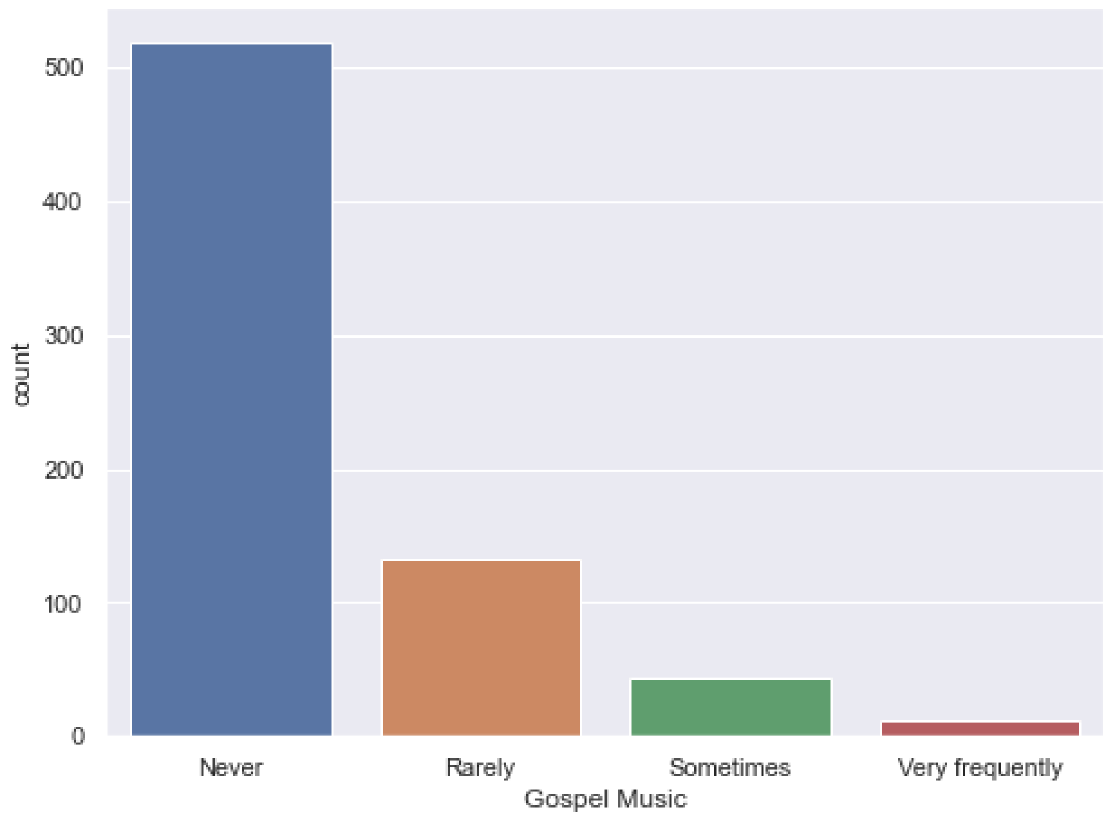

[23]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [EDM]'])
plt.xlabel('EDM Music')
```

[23]: `Text(0.5, 0, 'EDM Music')`



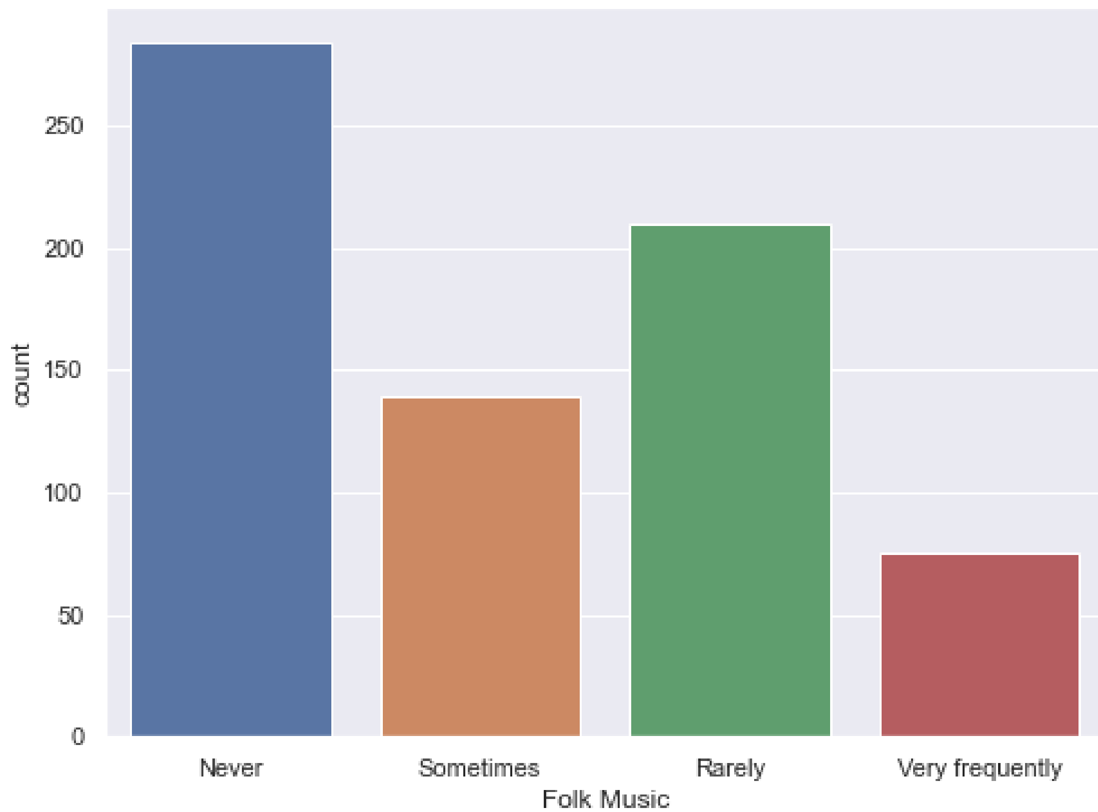

[24]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Gospel]'])
plt.xlabel('Gospel Music')
```

[24]: Text(0.5, 0, 'Gospel Music')



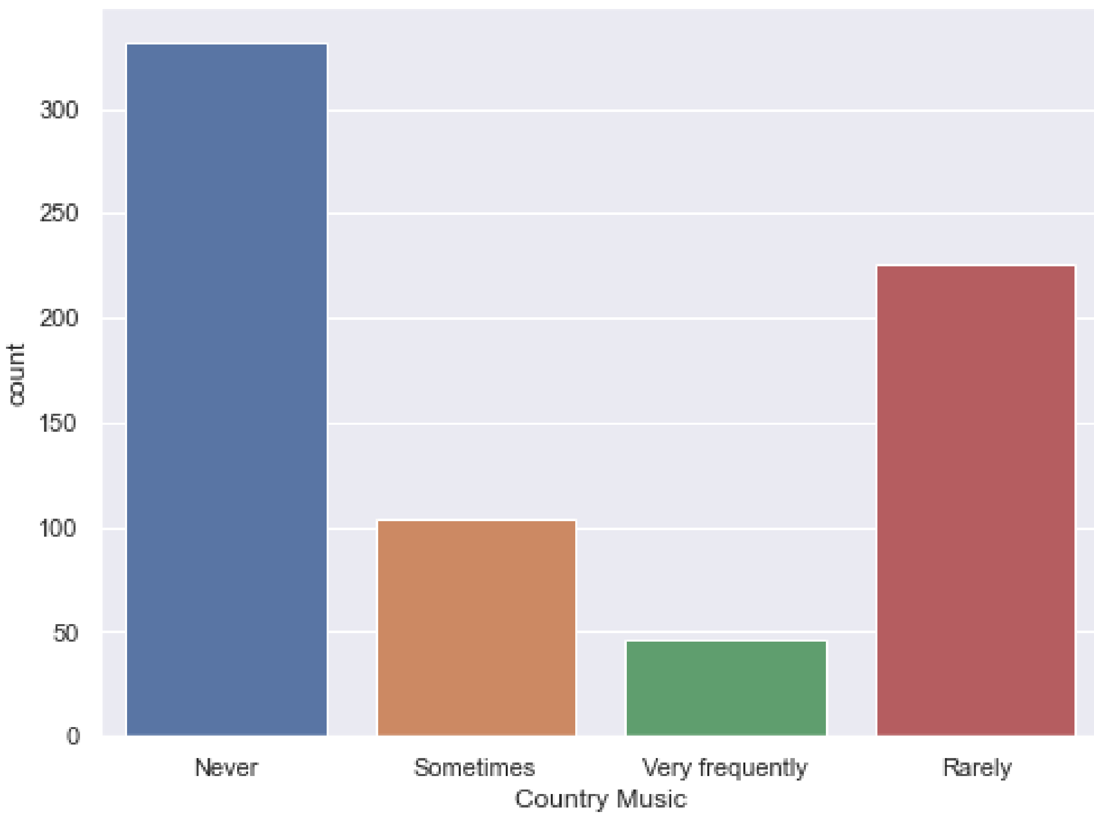

[25]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Folk]'])
plt.xlabel('Folk Music')
```

[25]: Text(0.5, 0, 'Folk Music')



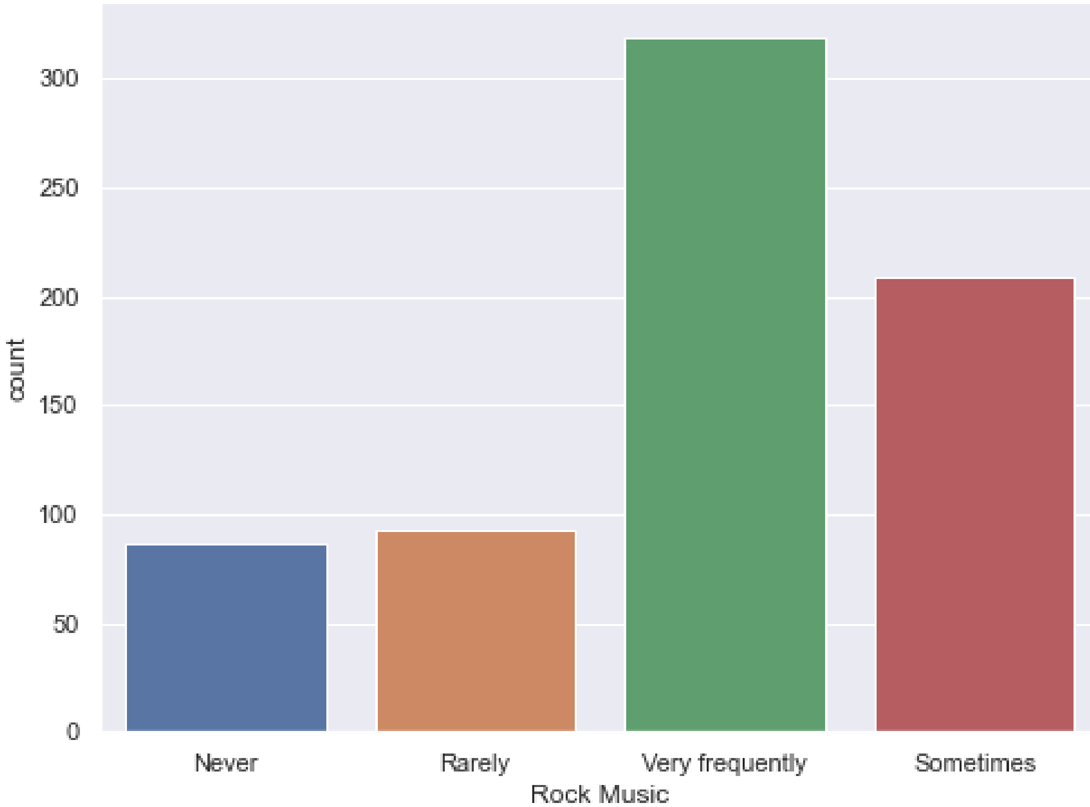

[26]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Country]'])
plt.xlabel('Country Music')
```

[26]:
```
Text(0.5, 0, 'Country Music')
```



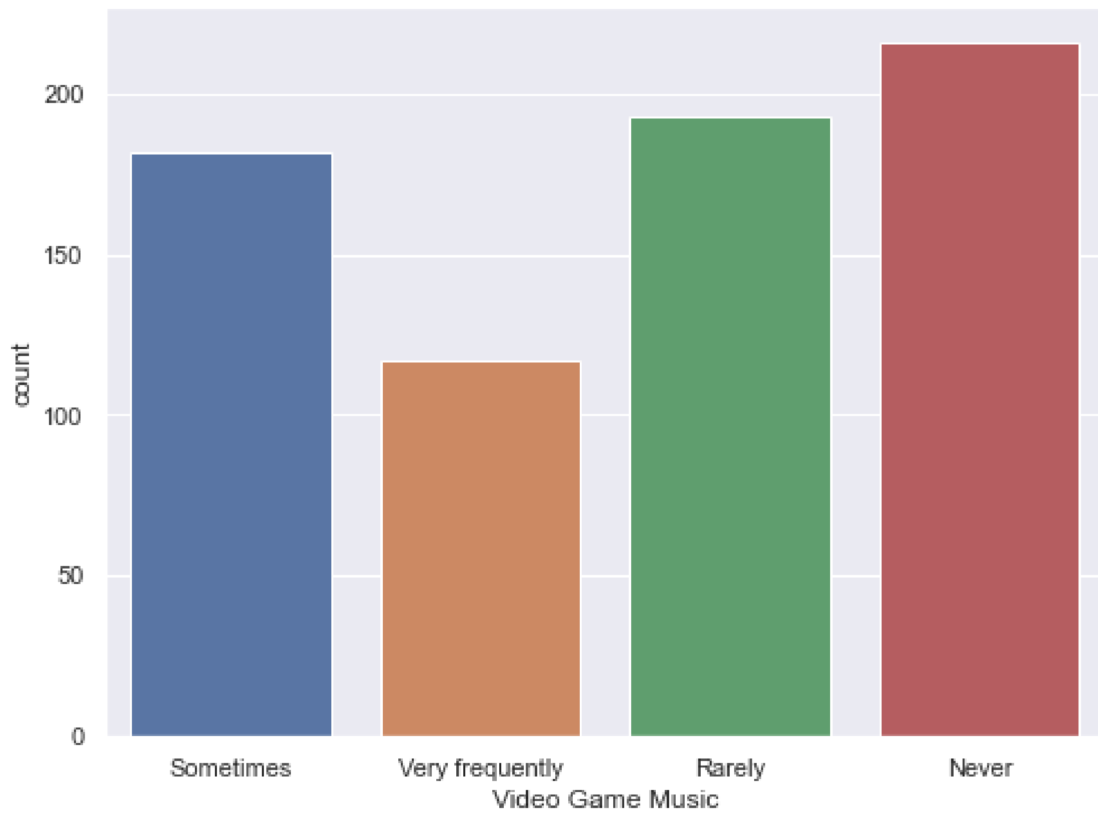

[27]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Rock]'])
plt.xlabel('Rock Music')
```

[27]: Text(0.5, 0, 'Rock Music')



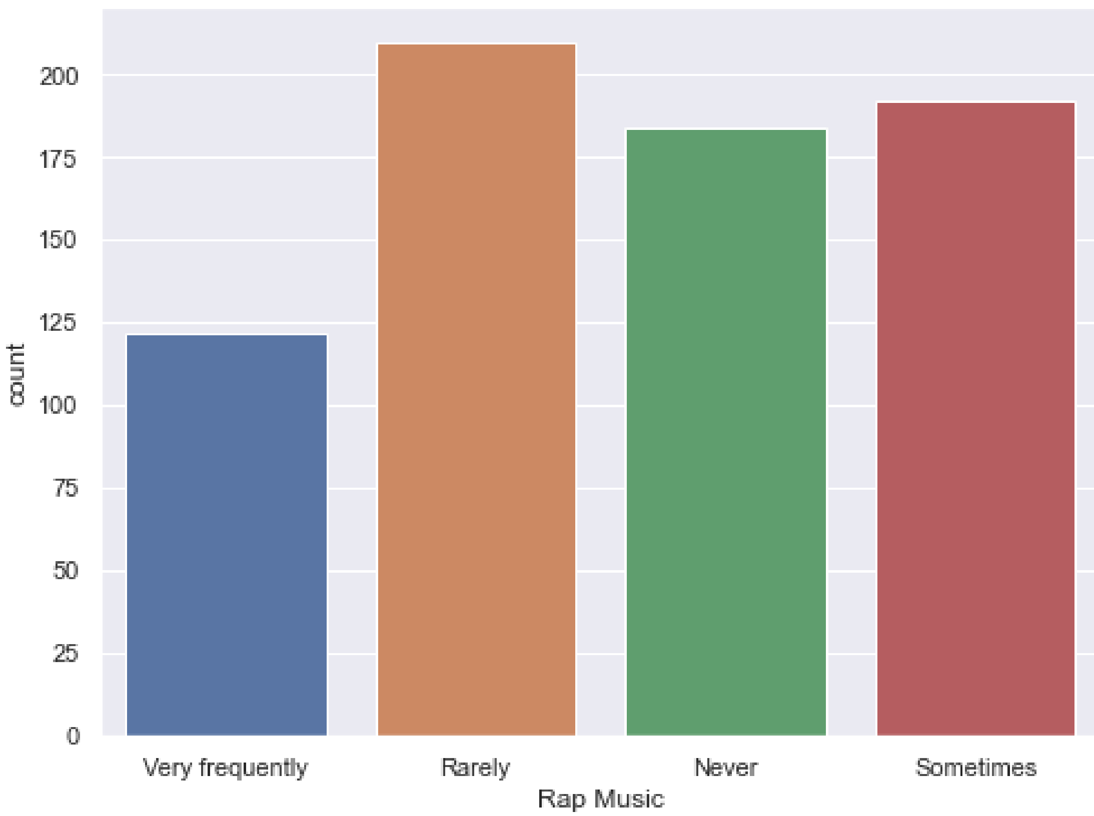

[28]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Video game music]'])
plt.xlabel('Video Game Music')
```

[28]: Text(0.5, 0, 'Video Game Music')



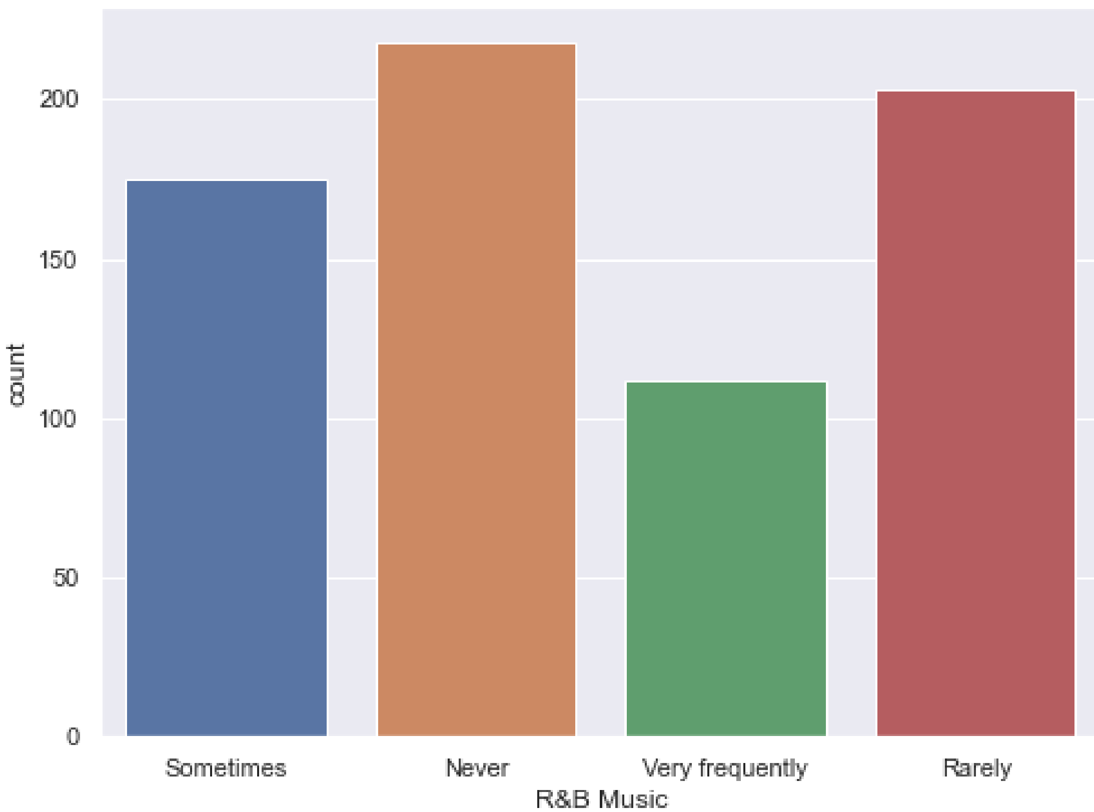

[29]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Rap]'])
plt.xlabel('Rap Music')
```

[29]: Text(0.5, 0, 'Rap Music')



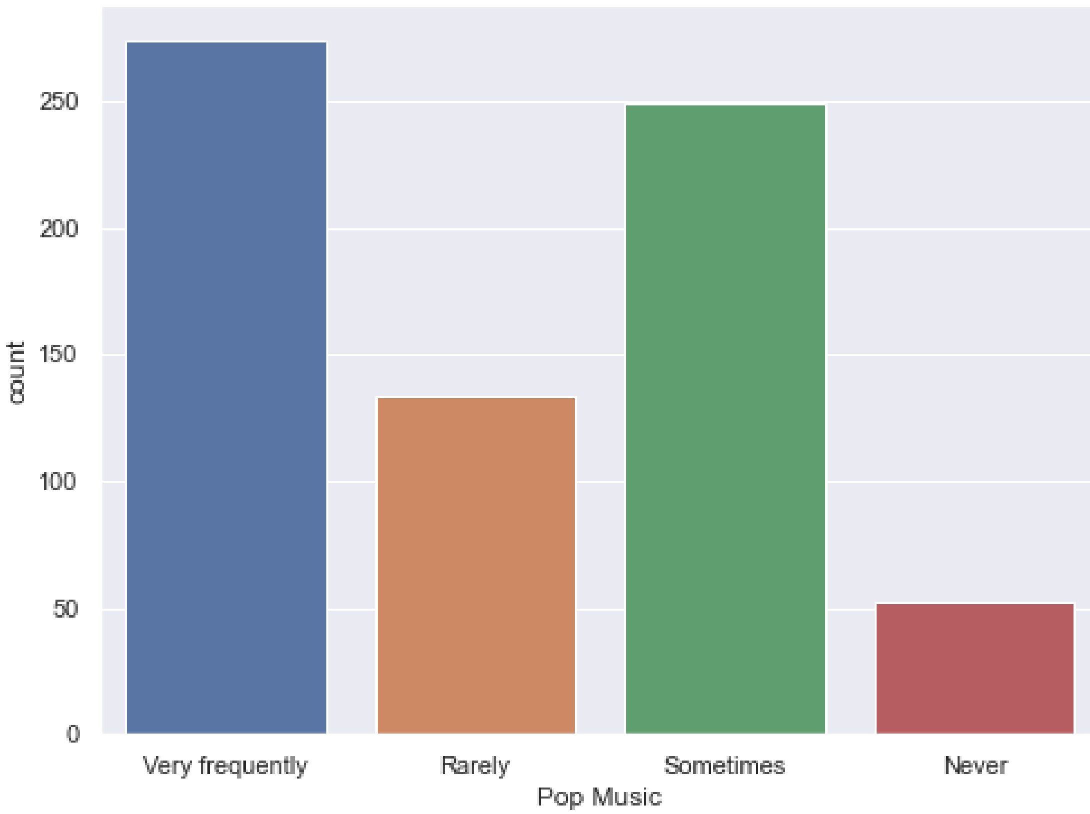

```
[30]: plt.figure(figsize=(8,6))
      seaborn.countplot(x=data['Frequency [R&B]'])
      plt.xlabel('R&B Music')
```

```
[30]: Text(0.5, 0, 'R&B Music')
```



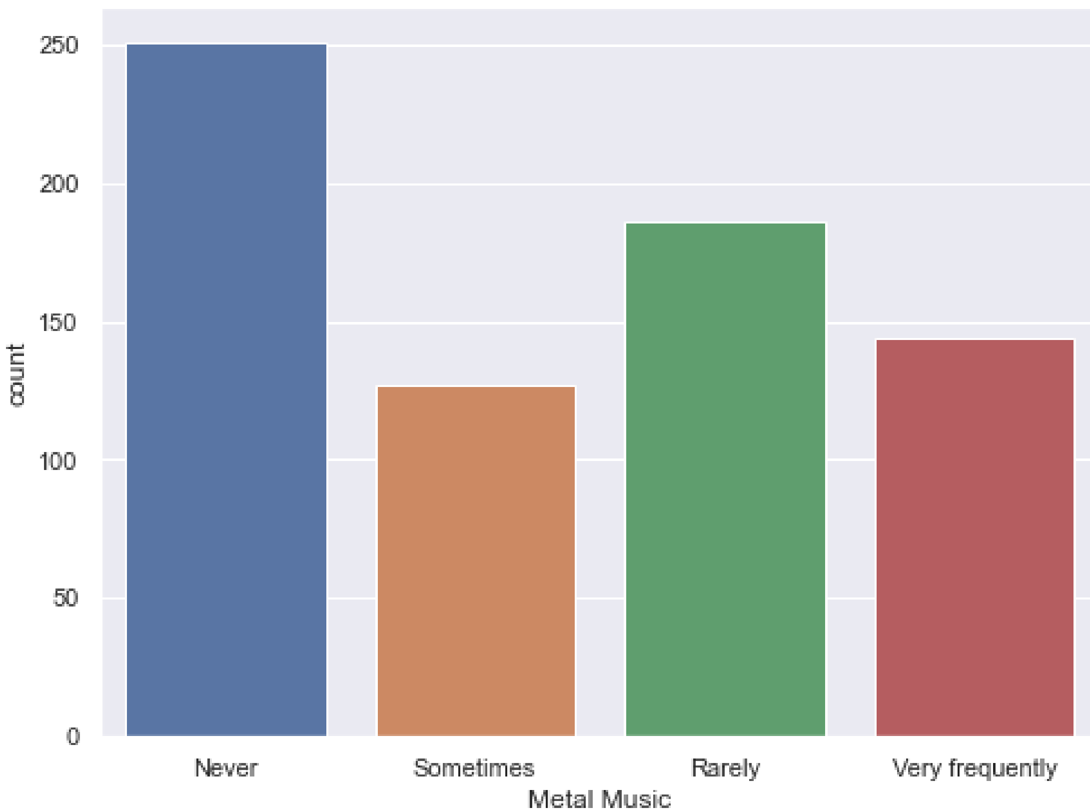

[31]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Pop]'])
plt.xlabel('Pop Music')
```

[31]:
```
Text(0.5, 0, 'Pop Music')
```



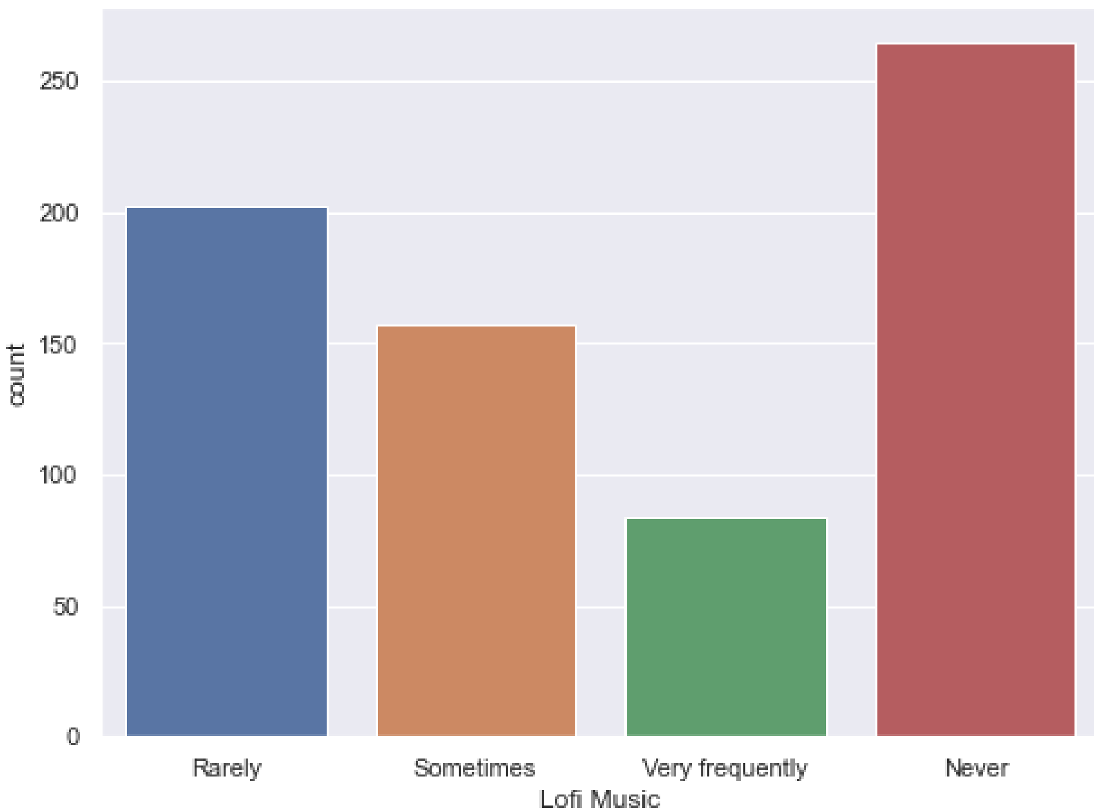

[32]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Metal]'])
plt.xlabel('Metal Music')
```

[32]: `Text(0.5, 0, 'Metal Music')`



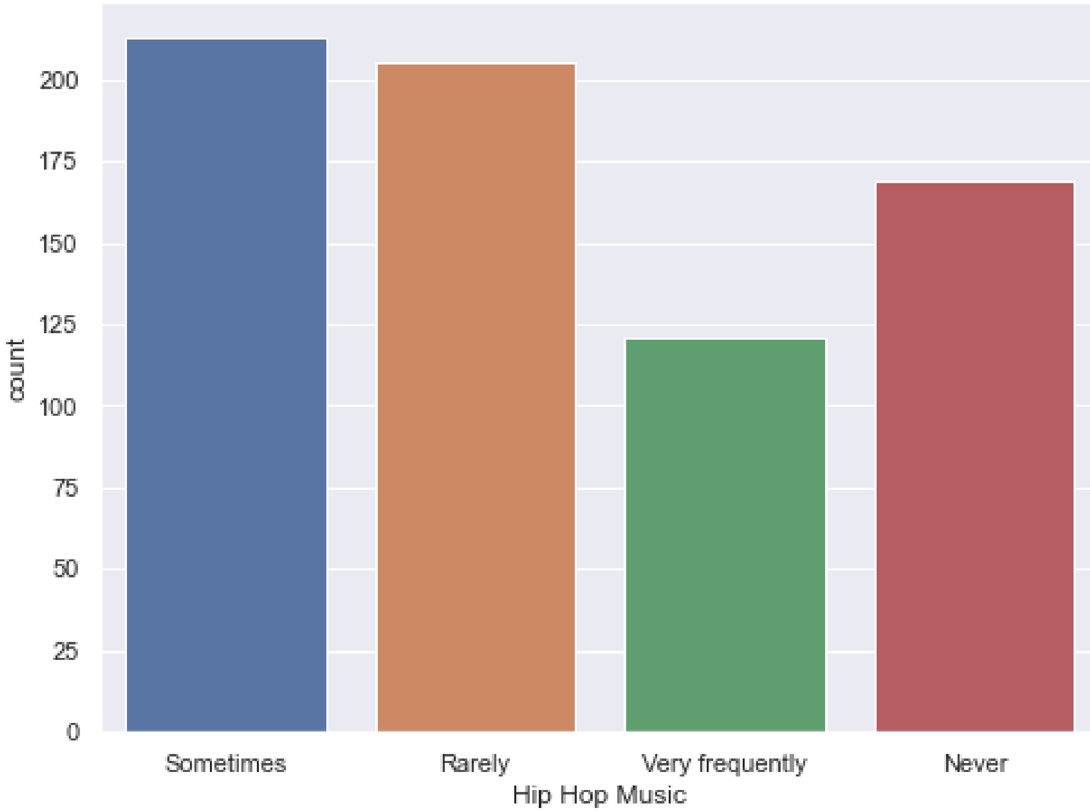

[33]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Lofi]'])
plt.xlabel('Lofi Music')
```

[33]: Text(0.5, 0, 'Lofi Music')



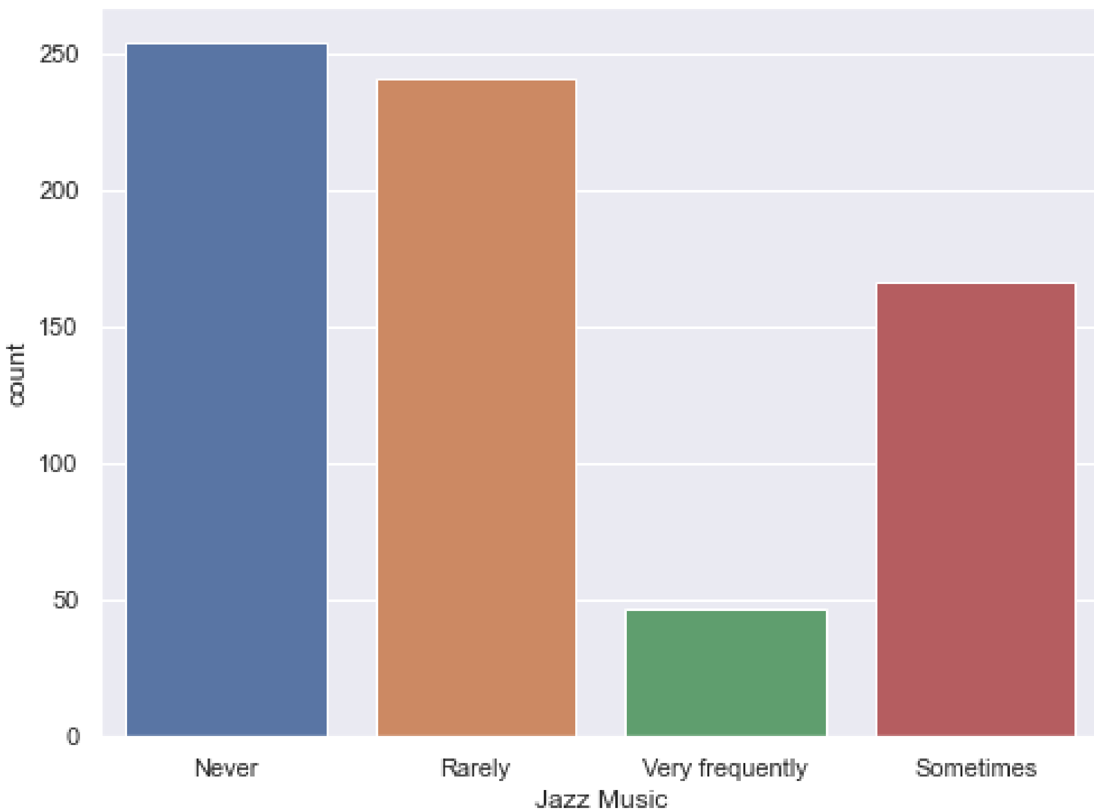

[34]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Hip hop]'])
plt.xlabel('Hip Hop Music')
```

[34]: `Text(0.5, 0, 'Hip Hop Music')`



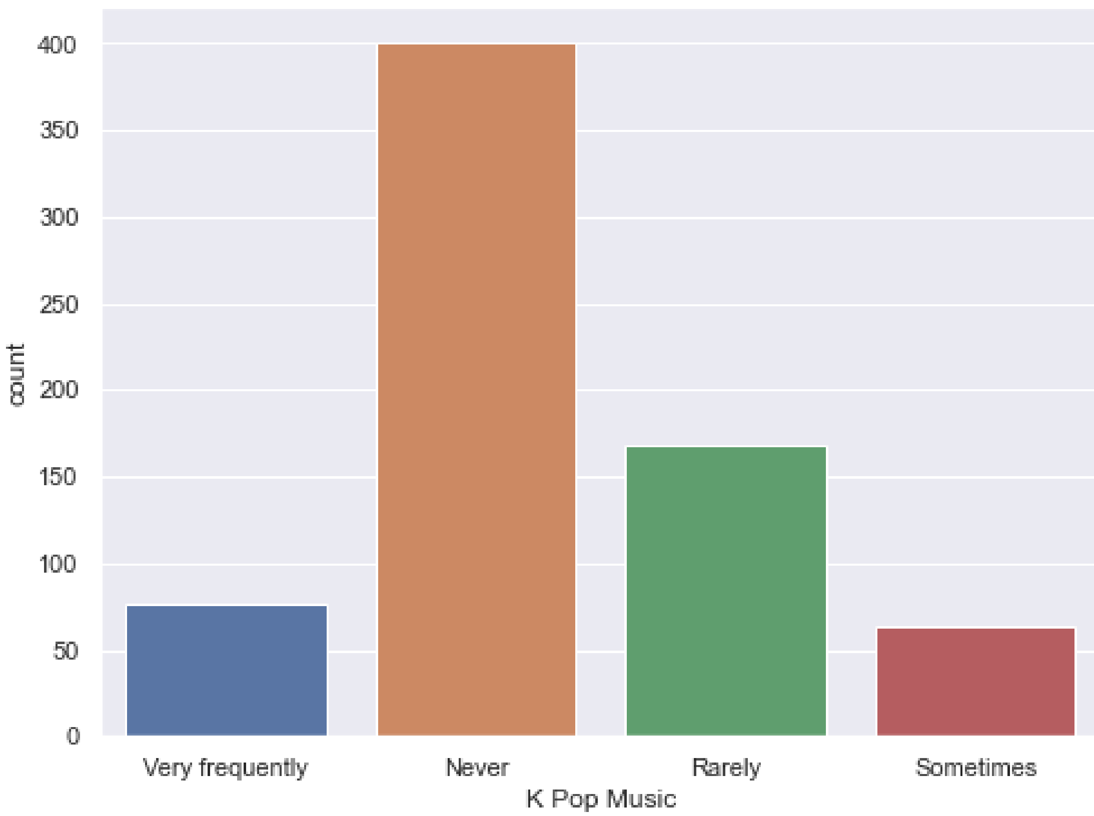

[35]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Jazz]'])
plt.xlabel('Jazz Music')
```

[35]: Text(0.5, 0, 'Jazz Music')



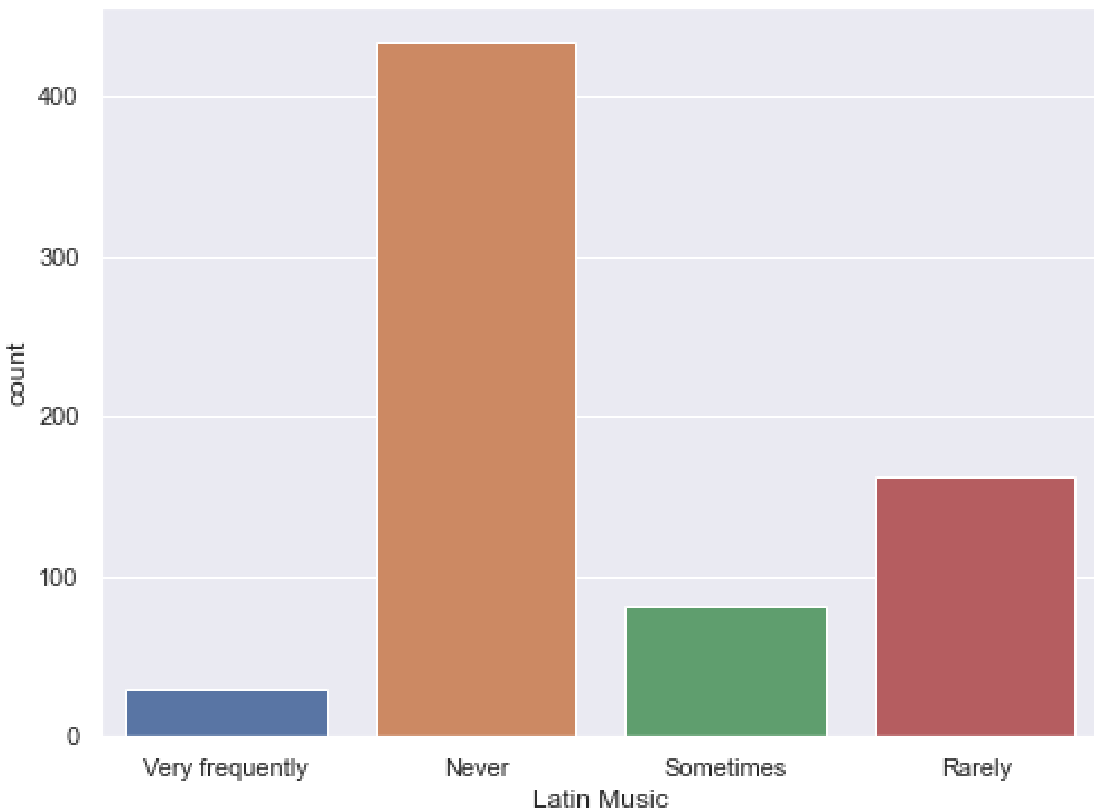

[36]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [K pop]'])
plt.xlabel('K Pop Music')
```

[36]: Text(0.5, 0, 'K Pop Music')




[37]:
```
plt.figure(figsize=(8,6))
seaborn.countplot(x=data['Frequency [Latin]'])
plt.xlabel('Latin Music')
```

[37]: Text(0.5, 0, 'Latin Music')



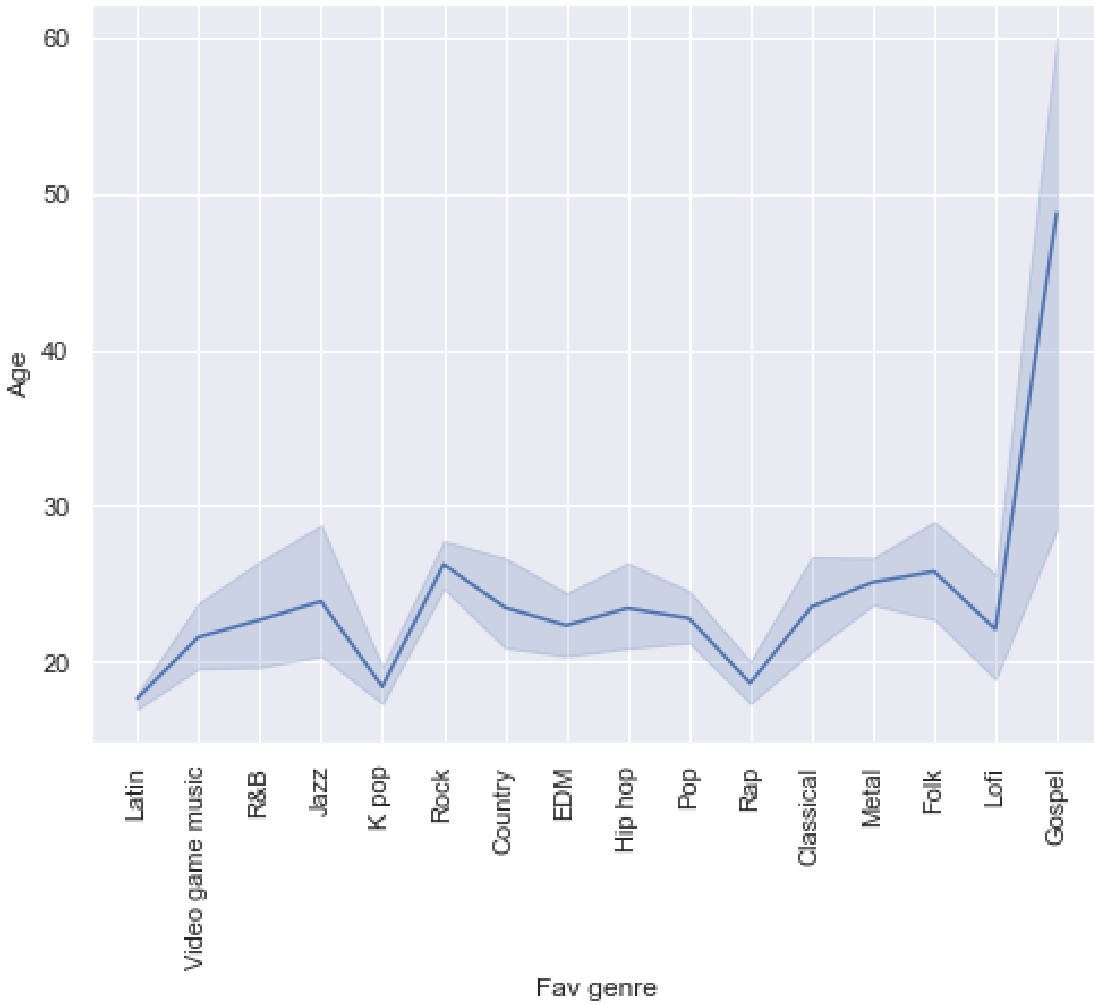

### 1.5 Bivariate Analysis

The bivariate analysis aims to uncover any potential correlations or associations between the mental health and the music someone is listening to.

[38]:
```
plt.figure(figsize=(8,6))
seaborn.lineplot(x=data['Fav genre'], y=data['Age'])
plt.xticks(rotation=90)
```

[38]:
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15],
 [Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
```

```
 Text(0, 0, ''),
 Text(0, 0, ''),
 Text(0, 0, ''),
 Text(0, 0, ''),
 Text(0, 0, '')])
```

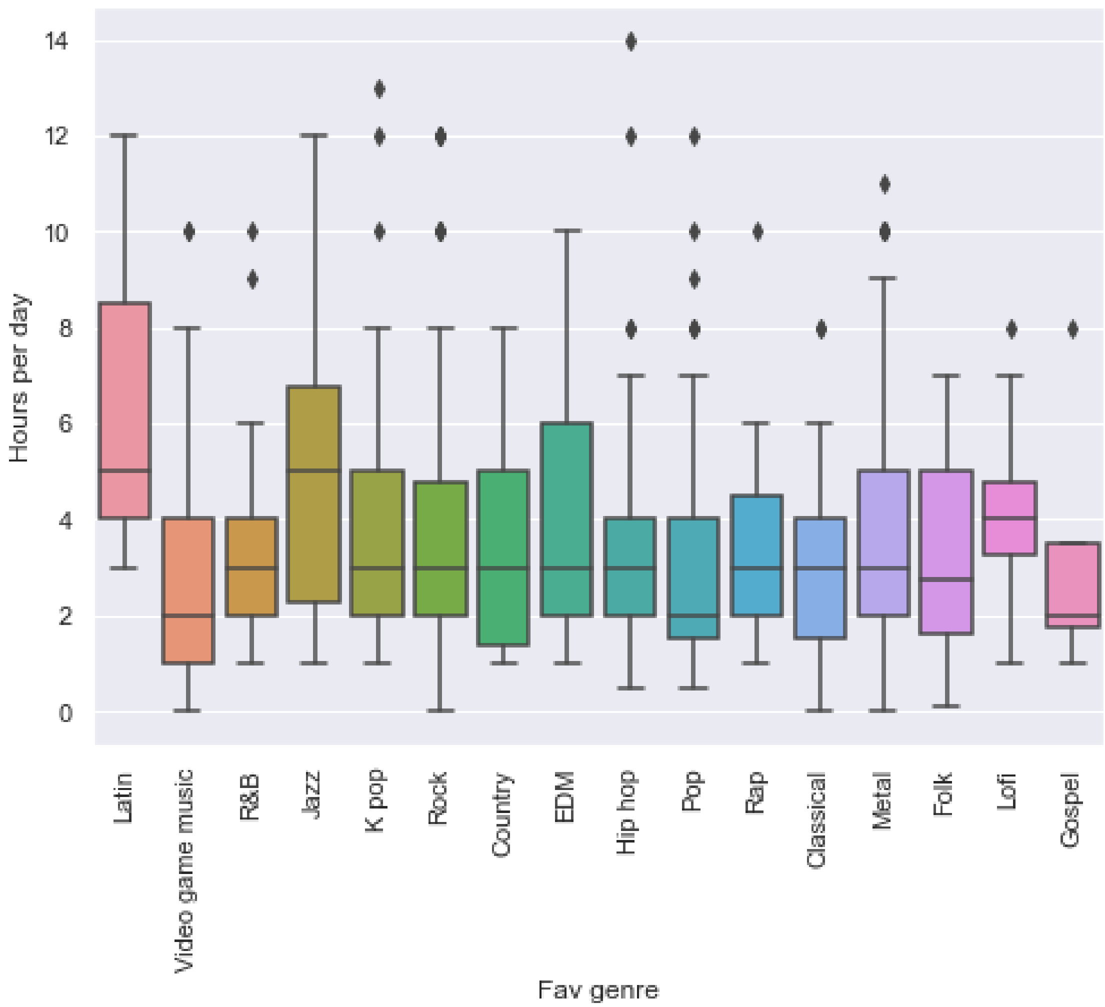



Individuals in their twenties tend to show a keen interest in *Latin*, *K-pop*, and *Rap* music genres, while *Gospel* music tends to be more popular among individuals aged 50 and above. Additionally, the aforementioned genres are commonly enjoyed by people ranging from 20 to 30 years old.

[39]:
```
plt.figure(figsize=(8,6))
seaborn.boxplot(x=data['Fav genre'], y=data['Hours per day'])
plt.xticks(rotation=90)
```

[39]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15]),
 [Text(0, 0, 'Latin'),
```

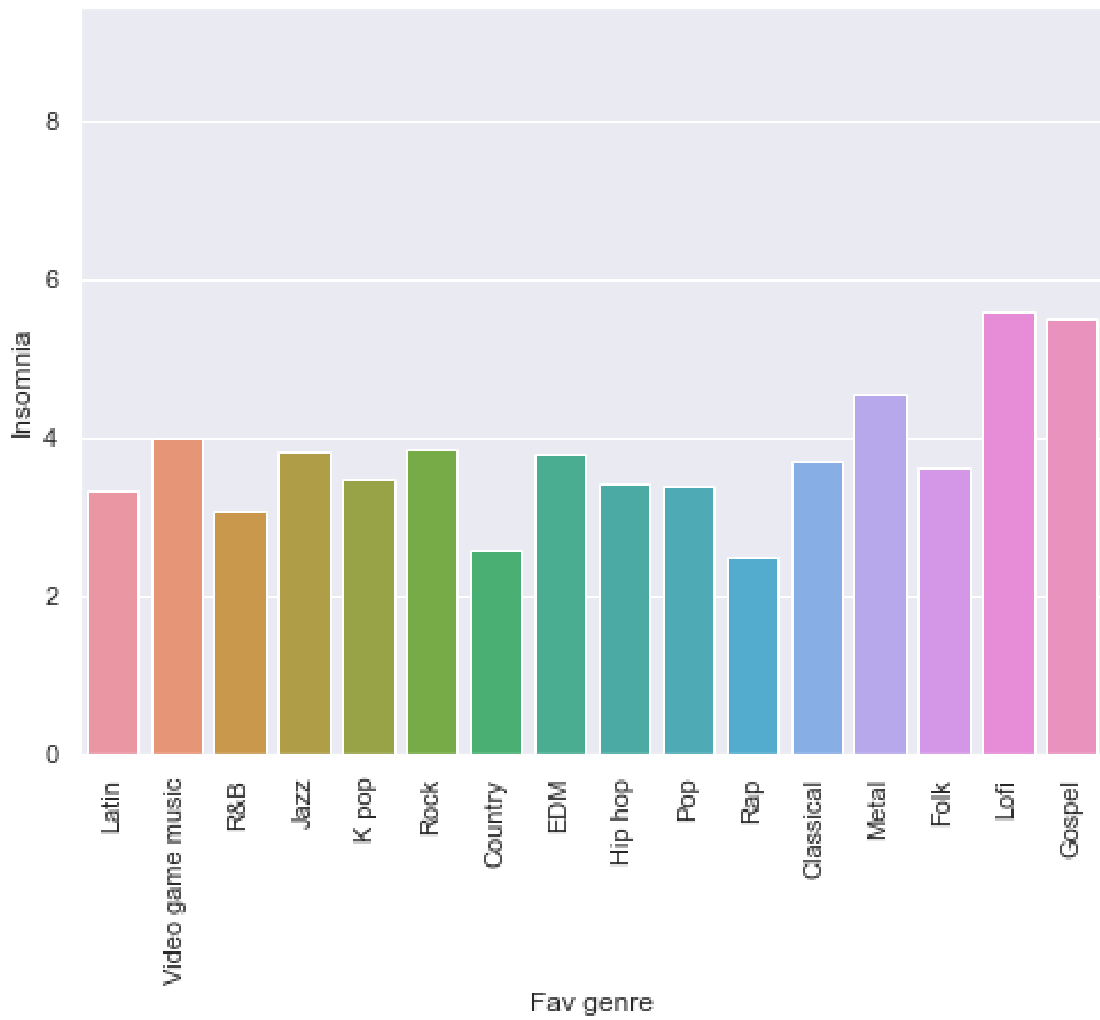
```
Text(1, 0, 'Video game music'),
Text(2, 0, 'R&B'),
Text(3, 0, 'Jazz'),
Text(4, 0, 'K pop'),
Text(5, 0, 'Rock'),
Text(6, 0, 'Country'),
Text(7, 0, 'EDM'),
Text(8, 0, 'Hip hop'),
Text(9, 0, 'Pop'),
Text(10, 0, 'Rap'),
Text(11, 0, 'Classical'),
Text(12, 0, 'Metal'),
Text(13, 0, 'Folk'),
Text(14, 0, 'Lofi'),
Text(15, 0, 'Gospel')])
```



Individuals who engage in 12-hour workdays tend to exhibit a higher inclination towards listening to *Latin* and *Jazz* music genres. Conversely, individuals who prefer *Gospel* music genres typically work fewer hours compared to others.

[40]:
```
plt.figure(figsize=(8,6))
seaborn.barplot(x=data['Fav genre'], y=data['Insomnia'], errwidth=0)
plt.xticks(rotation=90)
```

[40]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15]),
 [Text(0, 0, 'Latin'),
  Text(1, 0, 'Video game music'),
  Text(2, 0, 'R&B'),
  Text(3, 0, 'Jazz'),
  Text(4, 0, 'K pop'),
  Text(5, 0, 'Rock'),
  Text(6, 0, 'Country'),
  Text(7, 0, 'EDM'),
  Text(8, 0, 'Hip hop'),
  Text(9, 0, 'Pop'),
  Text(10, 0, 'Rap'),
  Text(11, 0, 'Classical'),
  Text(12, 0, 'Metal'),
  Text(13, 0, 'Folk'),
  Text(14, 0, 'Lofi'),
  Text(15, 0, 'Gospel')])
```

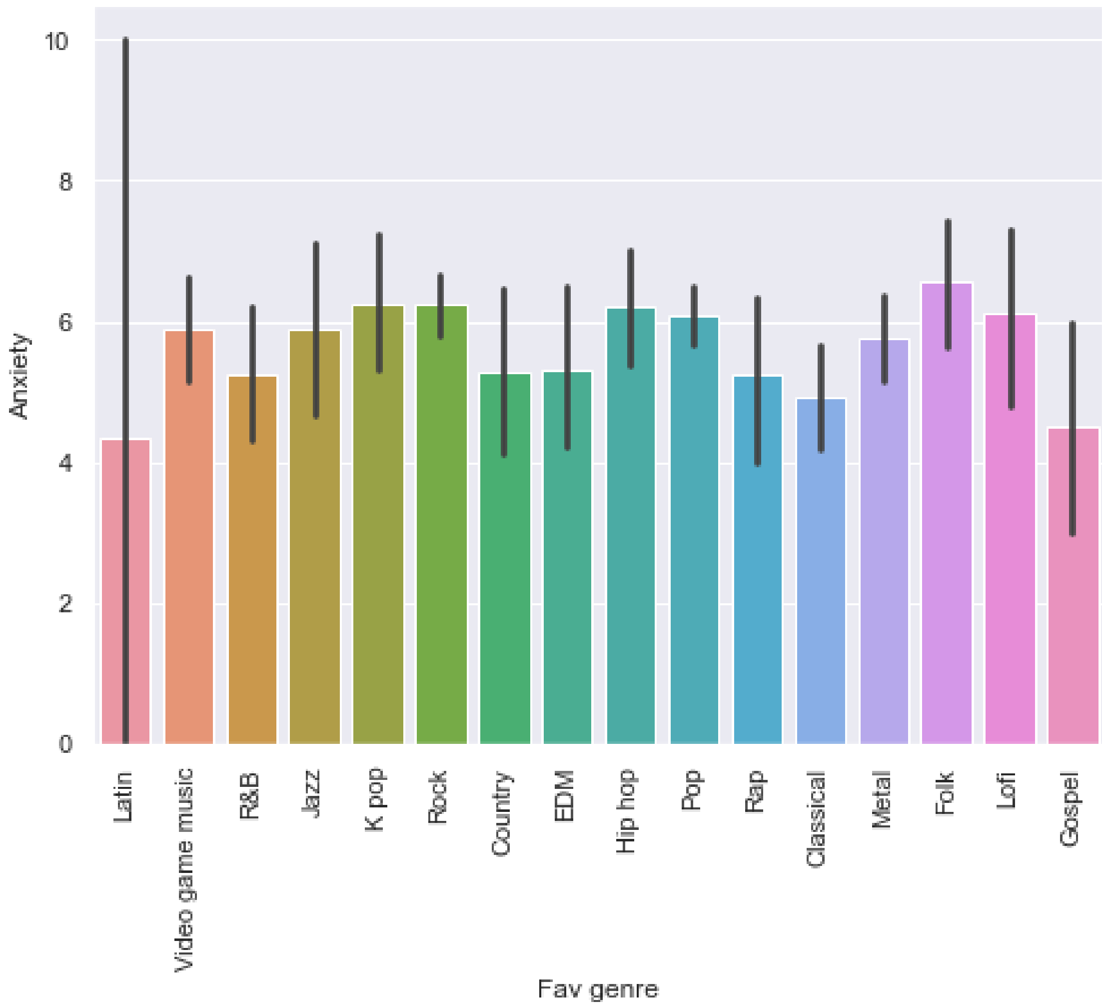



With the exception of *Metal*, *Lofi*, and *Gospel* genres, every listener typically has an insomnia level below 4.

[41]:
```
plt.figure(figsize=(8,6))
seaborn.barplot(x=data['Fav genre'], y=data['Anxiety'])
plt.xticks(rotation=90)
```

[41]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15]),
 [Text(0, 0, 'Latin'),
  Text(1, 0, 'Video game music'),
  Text(2, 0, 'R&B'),
  Text(3, 0, 'Jazz'),
  Text(4, 0, 'K pop'),
  Text(5, 0, 'Rock'),
  Text(6, 0, 'Country'),
```

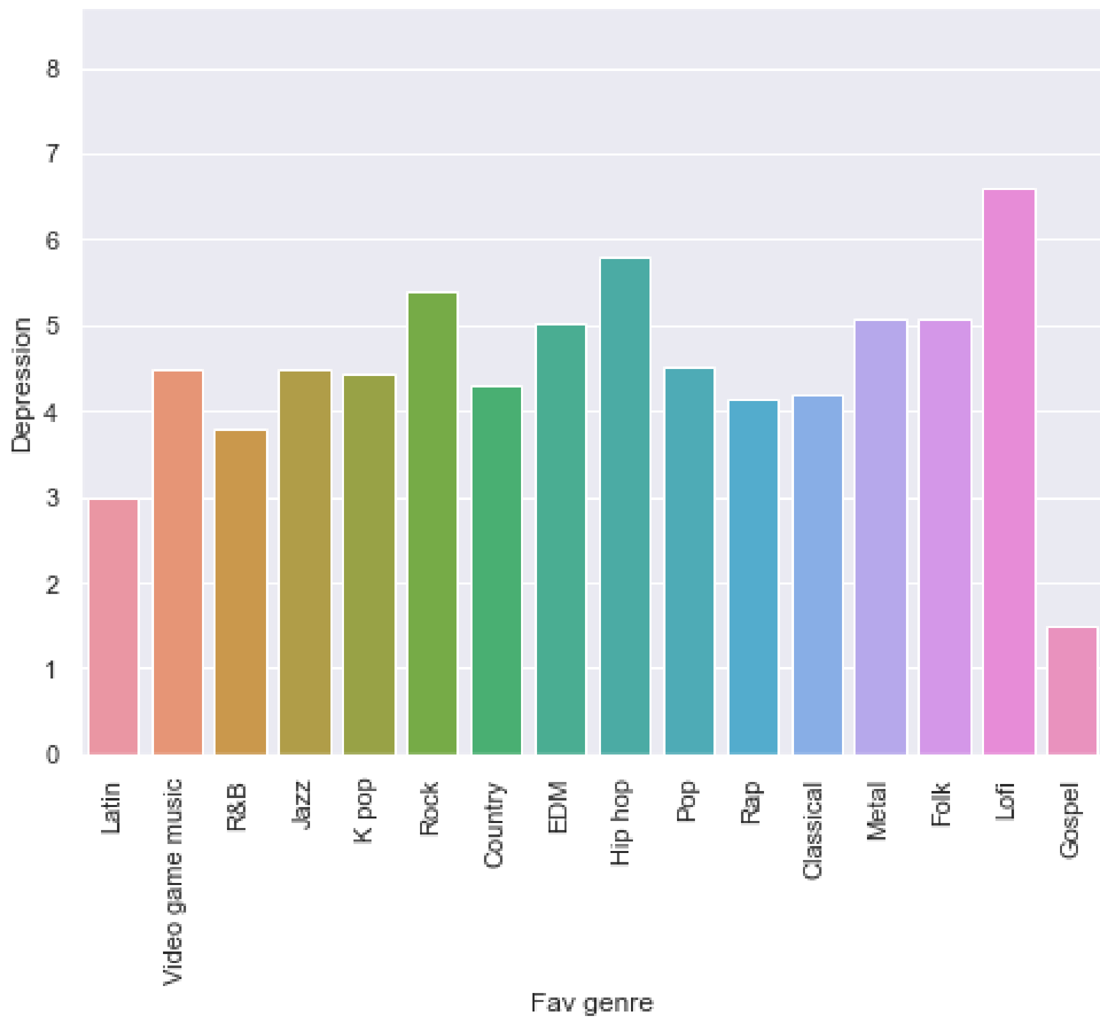
```
Text(7, 0, 'EDM'),
Text(8, 0, 'Hip hop'),
Text(9, 0, 'Pop'),
Text(10, 0, 'Rap'),
Text(11, 0, 'Classical'),
Text(12, 0, 'Metal'),
Text(13, 0, 'Folk'),
Text(14, 0, 'Lofi'),
Text(15, 0, 'Gospel')])
```




With the exception of *Rock*, *Jazz*, *K-pop*, *Hip hop*, *Pop*, and *Folk* music listeners, every listener generally has an anxiety level above 4. However, those who prefer the aforementioned genres tend to have anxiety levels above 6.

[42]:
```
plt.figure(figsize=(8,6))
seaborn.barplot(x=data['Fav genre'], y=data['Depression'], errwidth=0)
plt.xticks(rotation=90)
```

[42]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15]),
 [Text(0, 0, 'Latin'),
  Text(1, 0, 'Video game music'),
  Text(2, 0, 'R&B'),
  Text(3, 0, 'Jazz'),
  Text(4, 0, 'K pop'),
  Text(5, 0, 'Rock'),
  Text(6, 0, 'Country'),
  Text(7, 0, 'EDM'),
  Text(8, 0, 'Hip hop'),
  Text(9, 0, 'Pop'),
  Text(10, 0, 'Rap'),
  Text(11, 0, 'Classical'),
  Text(12, 0, 'Metal'),
  Text(13, 0, 'Folk'),
  Text(14, 0, 'Lofi'),
  Text(15, 0, 'Gospel')])
```

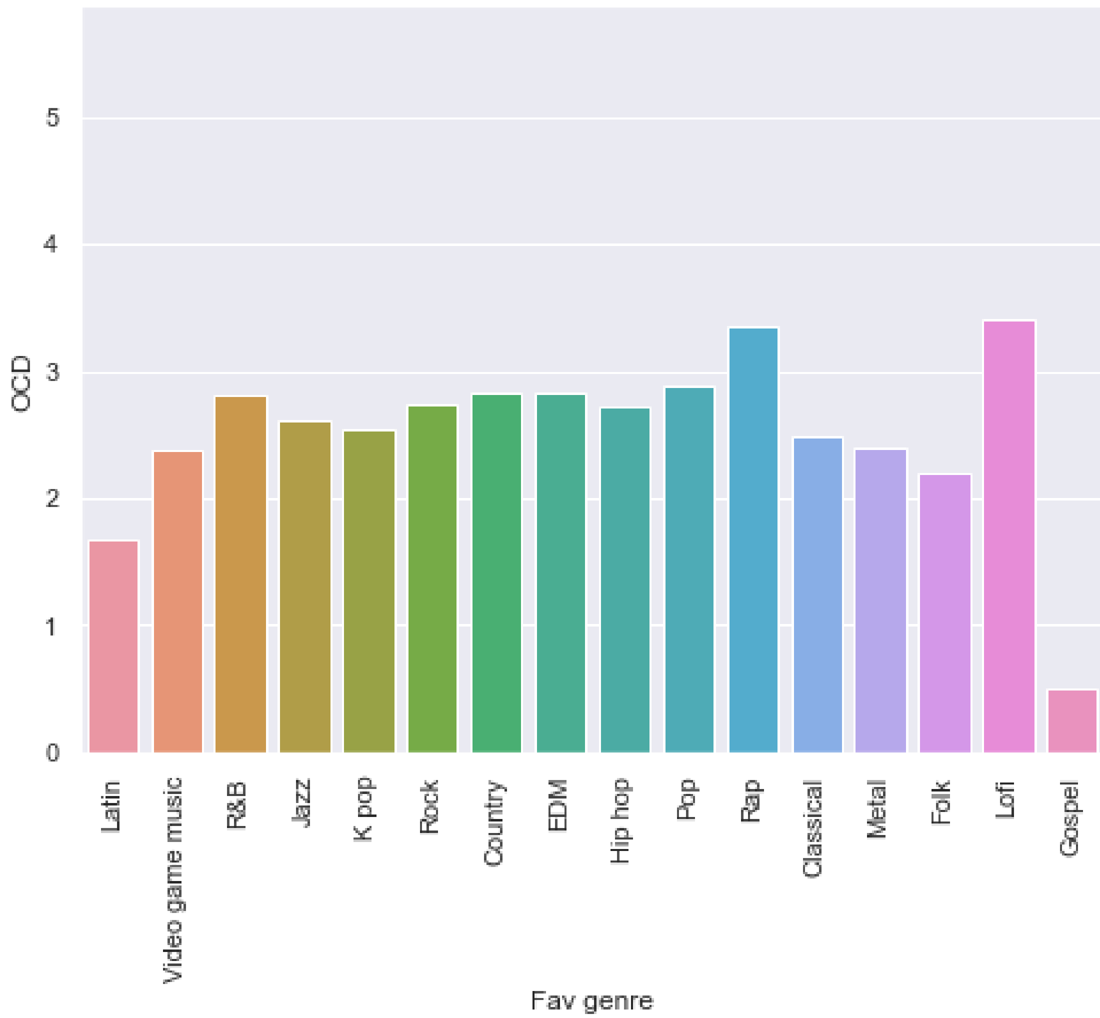



While every listener tends to have a depression level above 3, *Lofi*, *Hip hop*, and *Rock* music listeners specifically exhibit higher levels of depression, surpassing 5 on average.

[43]:
```
plt.figure(figsize=(8,6))
seaborn.barplot(x=data['Fav genre'], y=data['OCD'], errwidth=0)
plt.xticks(rotation=90)
```

[43]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15]),
 [Text(0, 0, 'Latin'),
  Text(1, 0, 'Video game music'),
  Text(2, 0, 'R&B'),
  Text(3, 0, 'Jazz'),
  Text(4, 0, 'K pop'),
  Text(5, 0, 'Rock'),
  Text(6, 0, 'Country'),
```

```
 Text(7, 0, 'EDM'),
 Text(8, 0, 'Hip hop'),
 Text(9, 0, 'Pop'),
 Text(10, 0, 'Rap'),
 Text(11, 0, 'Classical'),
 Text(12, 0, 'Metal'),
 Text(13, 0, 'Folk'),
 Text(14, 0, 'Lofi'),
 Text(15, 0, 'Gospel')])
```

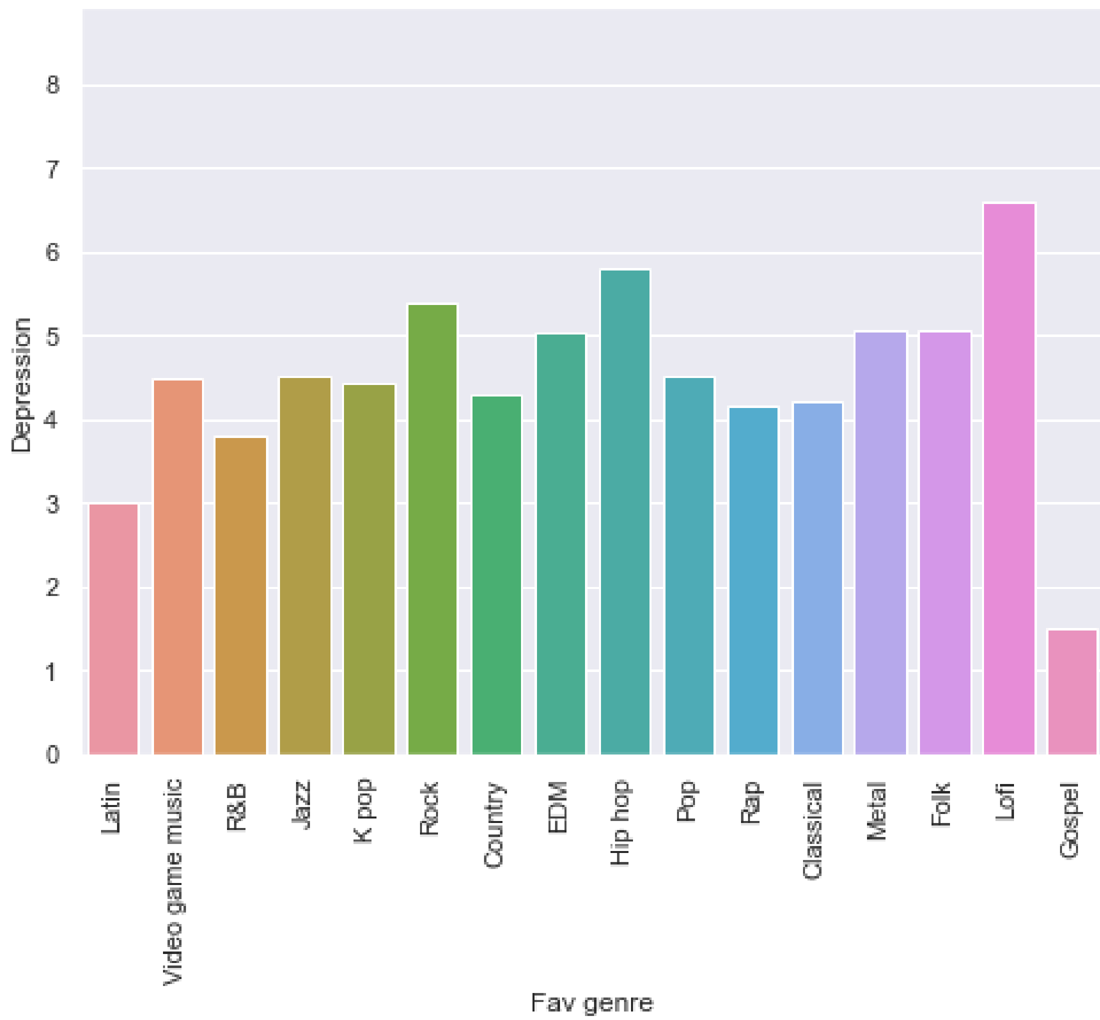



*Rap* and *Lofi* music listeners typically exhibit OCD levels above 3, indicating a higher prevalence of **obsessive-compulsive tendencies** in these groups.

```
[44]: plt.figure(figsize=(8,6))
      seaborn.barplot(x=data['Fav genre'], y=data['Depression'], errwidth=0)
```

```python
plt.xticks(rotation=90)
```

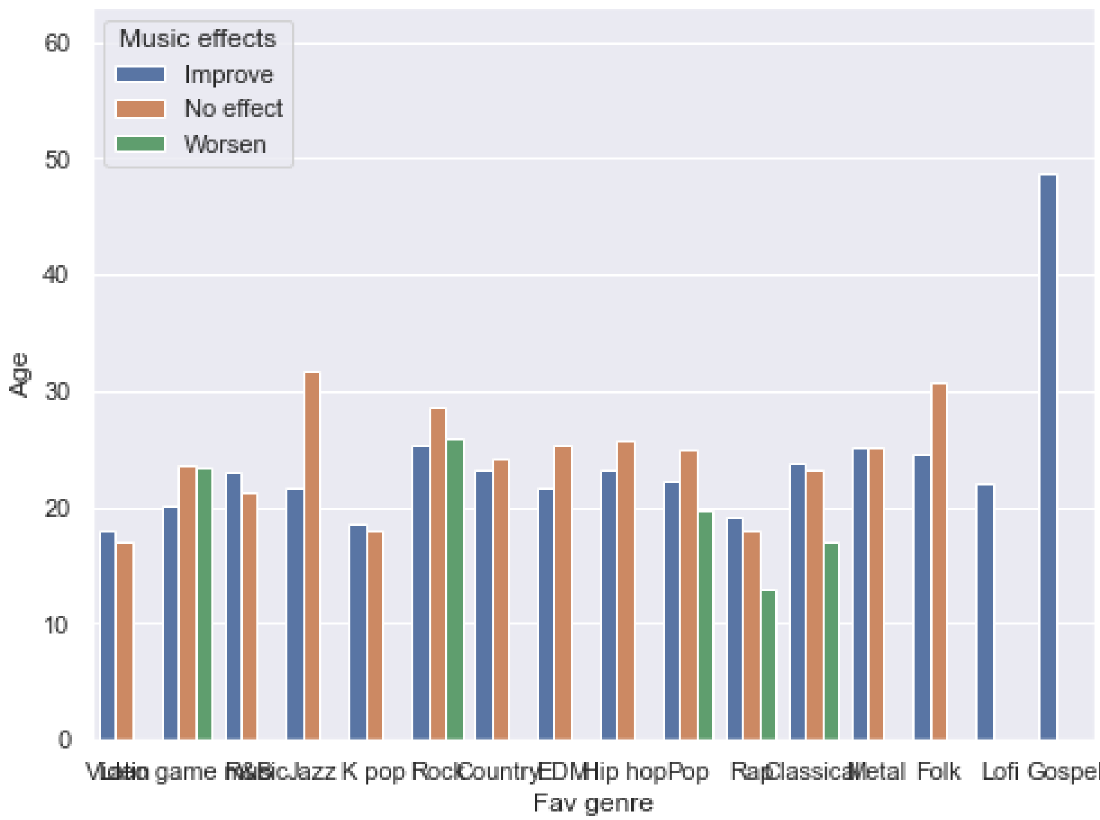
[44]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15]),
 [Text(0, 0, 'Latin'),
  Text(1, 0, 'Video game music'),
  Text(2, 0, 'R&B'),
  Text(3, 0, 'Jazz'),
  Text(4, 0, 'K pop'),
  Text(5, 0, 'Rock'),
  Text(6, 0, 'Country'),
  Text(7, 0, 'EDM'),
  Text(8, 0, 'Hip hop'),
  Text(9, 0, 'Pop'),
  Text(10, 0, 'Rap'),
  Text(11, 0, 'Classical'),
  Text(12, 0, 'Metal'),
  Text(13, 0, 'Folk'),
  Text(14, 0, 'Lofi'),
  Text(15, 0, 'Gospel')])
```




The *Lofi*, *Rock*, and *Hip Hop* listeners have Depression level above 5. While the rest are level 4 or below.

[45]:
```
plt.figure(figsize=(8,6))
seaborn.barplot(x=data['Fav genre'], y=data['Age'], hue=data['Music effects'],␣
↪errwidth=0)
```

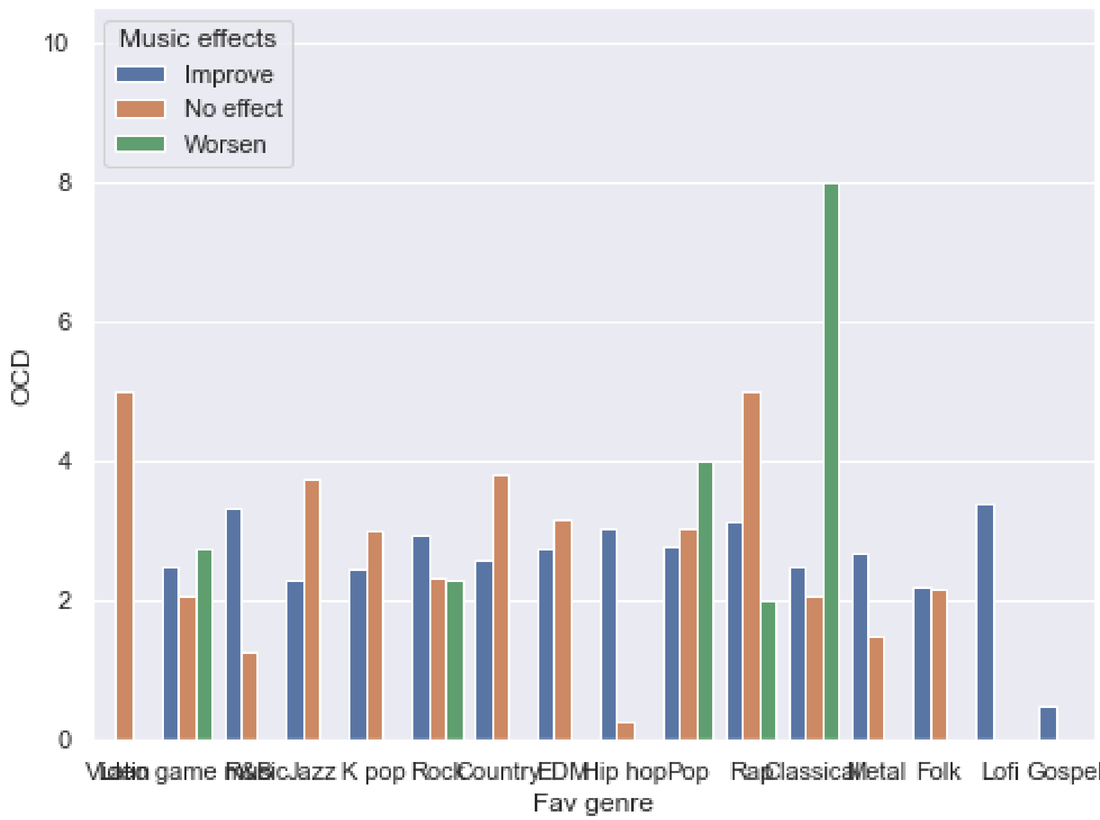
[45]: <AxesSubplot:xlabel='Fav genre', ylabel='Age'>




*Rock*, *video game music*, *pop*, *rap*, and *classical music* have worsened conditions, but listeners to music of all ages are improving in some way.

[46]:
```
plt.figure(figsize=(8,6))
seaborn.barplot(x=data['Fav genre'], y=data['OCD'], hue=data['Music effects'],
    errwidth=0)
```

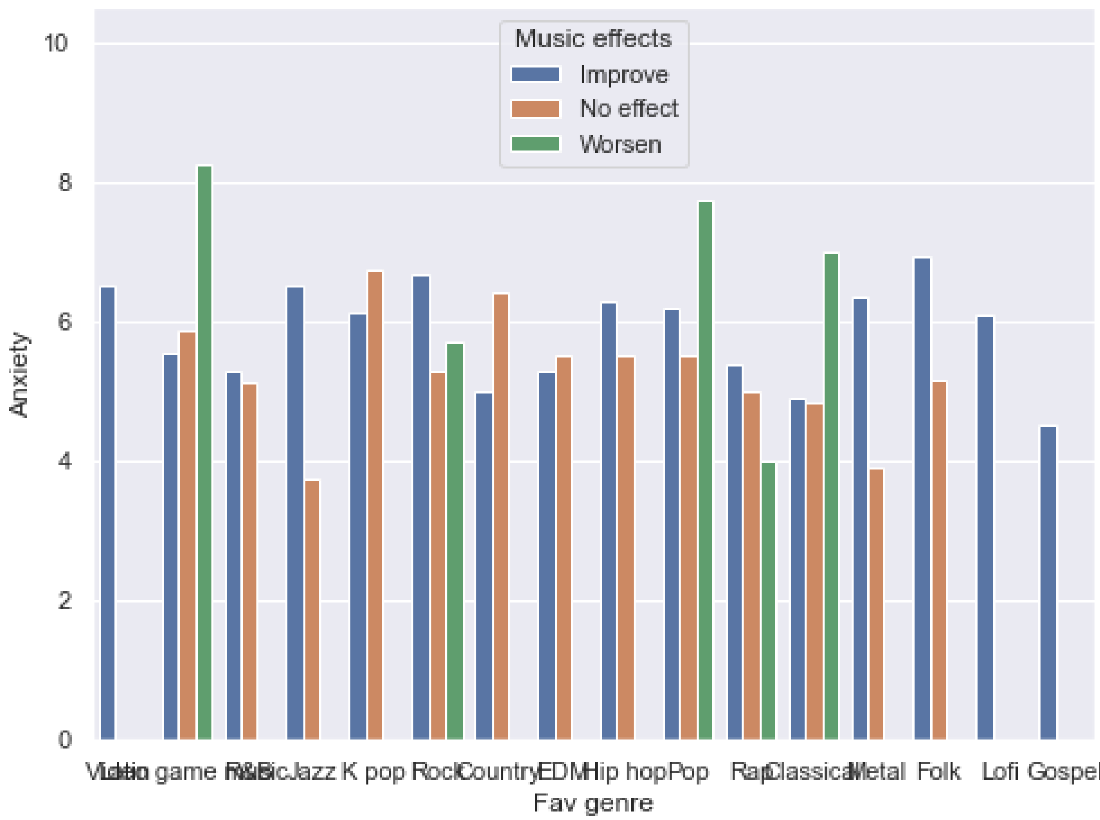
[46]: <AxesSubplot:xlabel='Fav genre', ylabel='OCD'>




People who listen to classical music have the greatest levels of **OCD**, followed by those who listen to *rock*, *video game music*, *pop*, *rap*, and other genres.

[47]:
```
plt.figure(figsize=(8,6))
seaborn.barplot(x=data['Fav genre'], y=data['Anxiety'], hue=data['Music␣
↪effects'], errwidth=0)
```

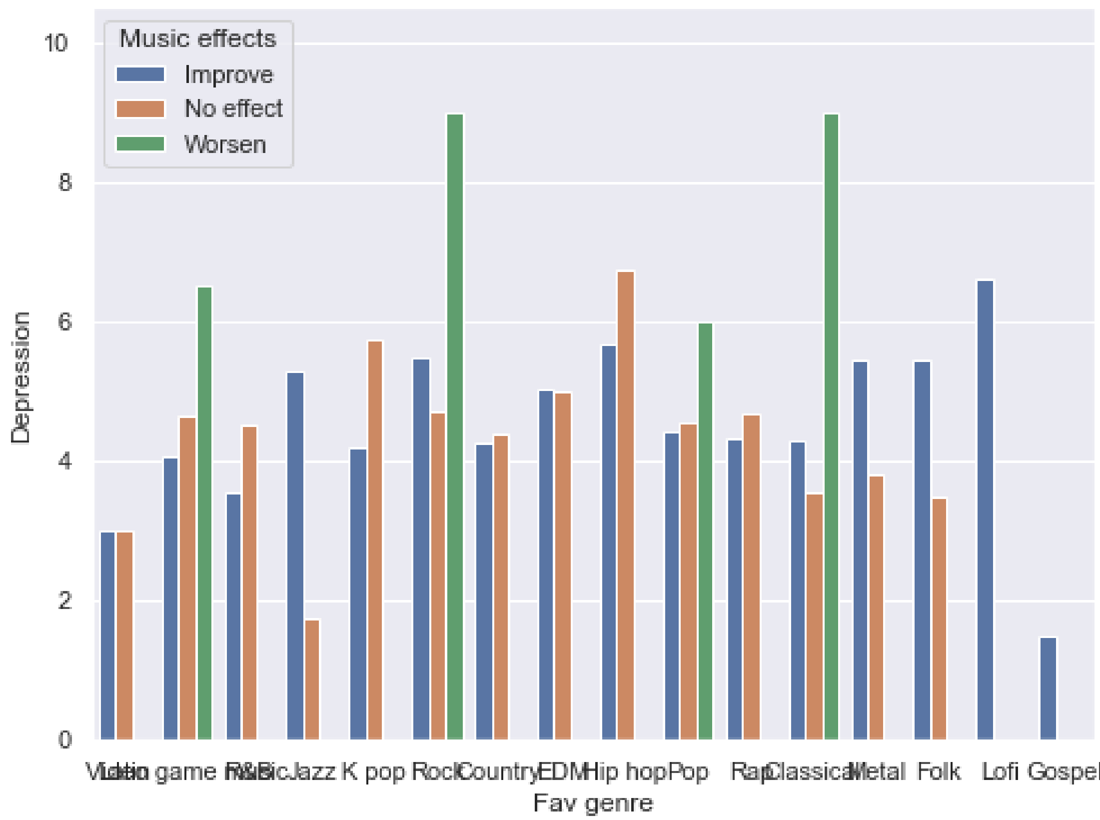
[47]: <AxesSubplot:xlabel='Fav genre', ylabel='Anxiety'>




Moreover, people who listen to *rock*, *video game music*, *pop*, *rap*, and *classical music* are generally anxious, while people who listen to *video game music* are the most anxious.

[48]:
```
plt.figure(figsize=(8,6))
seaborn.barplot(x=data['Fav genre'], y=data['Depression'], hue=data['Music␣
↪effects'], errwidth=0)
```

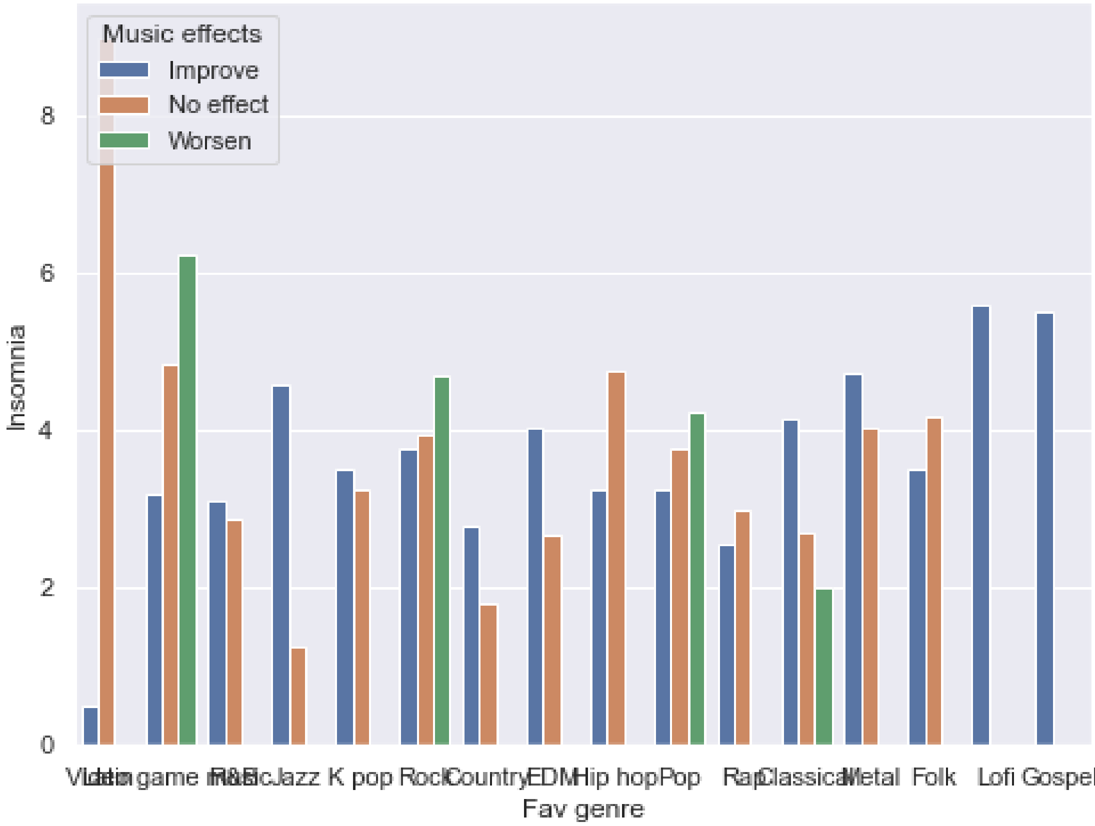
[48]:
```
<AxesSubplot:xlabel='Fav genre', ylabel='Depression'>
```




Clearly, people who listen to *rock*, *video game music*, *pop*, and *classical music* are more depressed.

[49]:
```
plt.figure(figsize=(8,6))
seaborn.barplot(x=data['Fav genre'], y=data['Insomnia'], hue=data['Music␣
↪effects'], errwidth=0)
```

[49]: `<AxesSubplot:xlabel='Fav genre', ylabel='Insomnia'>`




*Rock*, *video game music*, *pop*, and *classical music* listeners all experience some degree of **insomnia**, but **anxiety** is higher among *video game music* listeners.

### 1.6 Para terminar

This demonstrates the multiple qualities of music. Although it sometimes makes things worse, it could benefits our mental health.