AU1841095_Covid-19_Assignment

November 8, 2020

[1]:
```
!pip install pgmpy
```

```
Collecting pgmpy
  Downloading https://files.pythonhosted.org/packages/22/16/e2edff21fca7a2
cacaedd293caaaae5eaea832af0dd347c1159b4455be6b/pgmpy-0.1.12-py3-none-any.whl
(322kB)
     || 327kB 2.7MB/s
Requirement already satisfied: tqdm in /usr/local/lib/python3.6/dist-
packages (from pgmpy) (4.41.1)
Requirement already satisfied: pandas in /usr/local/lib/python3.6/dist-packages
(from pgmpy) (1.1.4)
Requirement already satisfied: numpy in /usr/local/lib/python3.6/dist-packages
(from pgmpy) (1.18.5)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.6/dist-
packages (from pgmpy) (0.22.2.post1)
Requirement already satisfied: networkx in /usr/local/lib/python3.6/dist-
packages (from pgmpy) (2.5)
Requirement already satisfied: torch in /usr/local/lib/python3.6/dist-packages
(from pgmpy) (1.7.0+cu101)
Requirement already satisfied: joblib in /usr/local/lib/python3.6/dist-packages
(from pgmpy) (0.17.0)
Requirement already satisfied: statsmodels in /usr/local/lib/python3.6/dist-
packages (from pgmpy) (0.10.2)
Requirement already satisfied: scipy in /usr/local/lib/python3.6/dist-packages
(from pgmpy) (1.4.1)
Requirement already satisfied: pyparsing in /usr/local/lib/python3.6/dist-
packages (from pgmpy) (2.4.7)
Requirement already satisfied: python-dateutil>=2.7.3 in
/usr/local/lib/python3.6/dist-packages (from pandas->pgmpy) (2.8.1)
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.6/dist-
packages (from pandas->pgmpy) (2018.9)
Requirement already satisfied: decorator>=4.3.0 in /usr/local/lib/python3.6
/dist-packages (from networkx->pgmpy) (4.4.2)
Requirement already satisfied: dataclasses in /usr/local/lib/python3.6/dist-
packages (from torch->pgmpy) (0.7)
Requirement already satisfied: future in /usr/local/lib/python3.6/dist-packages
(from torch->pgmpy) (0.16.0)
```

```
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.6
/dist-packages (from torch->pgmpy) (3.7.4.3)
Requirement already satisfied: patsy>=0.4.0 in /usr/local/lib/python3.6/dist-
packages (from statsmodels->pgmpy) (0.5.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.6/dist-
packages (from python-dateutil>=2.7.3->pandas->pgmpy) (1.15.0)
Installing collected packages: pgmpy
Successfully installed pgmpy-0.1.12
```

[2]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

from pgmpy.models import BayesianModel
from pgmpy.estimators import MaximumLikelihoodEstimator
from pgmpy.inference import VariableElimination
import networkx as nx
```

```
/usr/local/lib/python3.6/dist-packages/statsmodels/tools/_testing.py:19:
FutureWarning: pandas.util.testing is deprecated. Use the functions in the
public API at pandas.testing instead.
  import pandas.util.testing as tm
```

[3]:
```
df = pd.read_csv("Cleaned-Data.csv")
df
```

[3]:

	Fever	Tiredness	Dry-Cough	...	Contact_No	Contact_Yes	Country
0	1	1	1	...	0	1	China
1	1	1	1	...	1	0	China
2	1	1	1	...	0	0	China
3	1	1	1	...	0	1	China
4	1	1	1	...	1	0	China
...	...	...	...	...	...	...	...
316795	0	0	0	...	1	0	Other
316796	0	0	0	...	0	0	Other
316797	0	0	0	...	0	1	Other
316798	0	0	0	...	1	0	Other
316799	0	0	0	...	0	0	Other

[316800 rows x 27 columns]

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 316800 entries, 0 to 316799
Data columns (total 27 columns):
 #   Column                  Non-Null Count   Dtype
```

```
---  ------                    --------------   -----
 0   Fever                     316800 non-null  int64
 1   Tiredness                 316800 non-null  int64
 2   Dry-Cough                 316800 non-null  int64
 3   Difficulty-in-Breathing   316800 non-null  int64
 4   Sore-Throat               316800 non-null  int64
 5   None_Sympton              316800 non-null  int64
 6   Pains                     316800 non-null  int64
 7   Nasal-Congestion          316800 non-null  int64
 8   Runny-Nose                316800 non-null  int64
 9   Diarrhea                  316800 non-null  int64
 10  None_Experiencing         316800 non-null  int64
 11  Age_0-9                   316800 non-null  int64
 12  Age_10-19                 316800 non-null  int64
 13  Age_20-24                 316800 non-null  int64
 14  Age_25-59                 316800 non-null  int64
 15  Age_60+                   316800 non-null  int64
 16  Gender_Female             316800 non-null  int64
 17  Gender_Male               316800 non-null  int64
 18  Gender_Transgender        316800 non-null  int64
 19  Severity_Mild             316800 non-null  int64
 20  Severity_Moderate         316800 non-null  int64
 21  Severity_None             316800 non-null  int64
 22  Severity_Severe           316800 non-null  int64
 23  Contact_Dont-Know         316800 non-null  int64
 24  Contact_No                316800 non-null  int64
 25  Contact_Yes               316800 non-null  int64
 26  Country                   316800 non-null  object
dtypes: int64(26), object(1)
memory usage: 65.3+ MB
```

[5]:
```
df.describe()
```

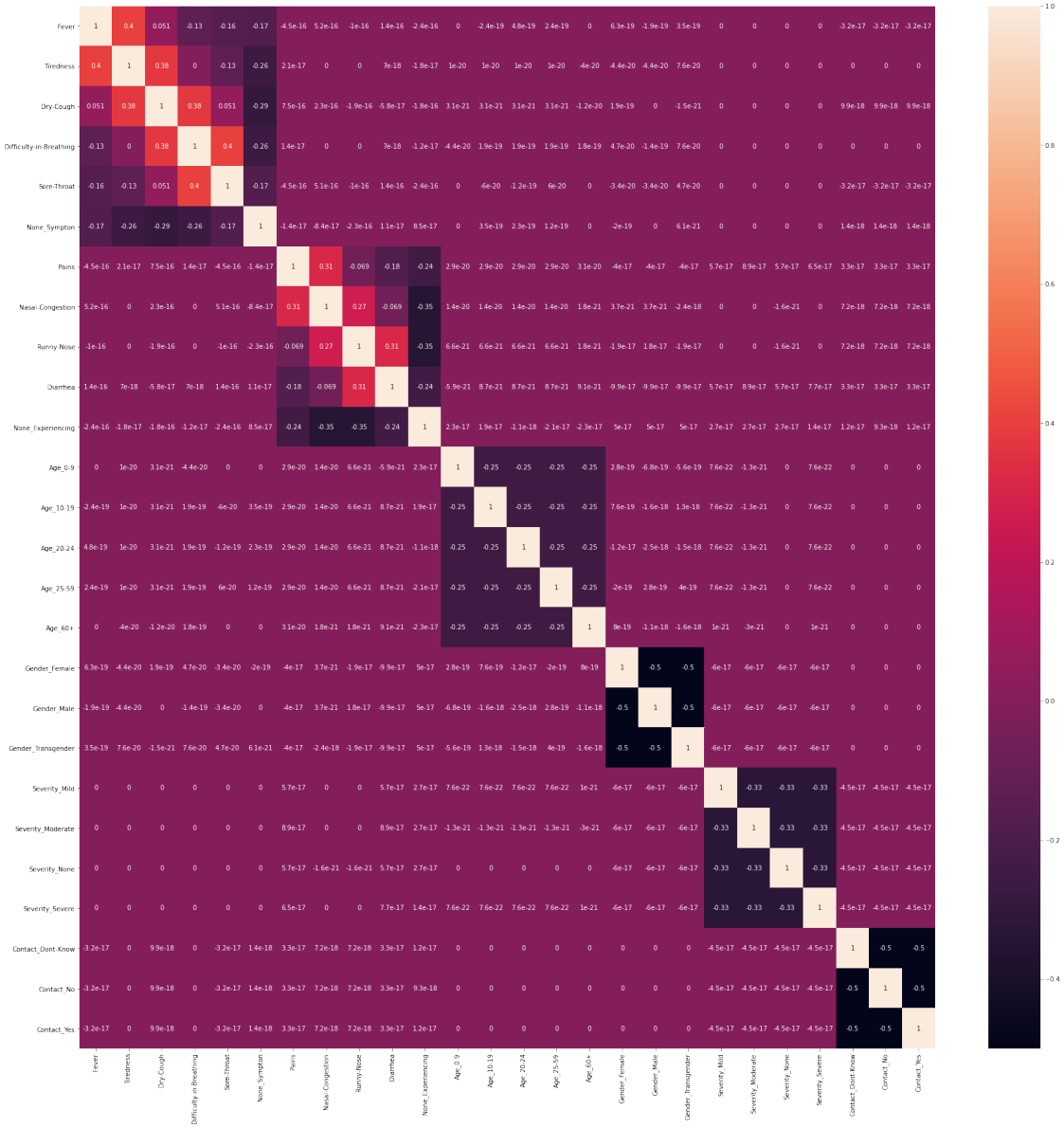
[5]:
```
              Fever      Tiredness  ...     Contact_No    Contact_Yes
count  316800.000000  316800.000000  ...  316800.000000  316800.000000
mean        0.312500       0.500000  ...       0.333333       0.333333
std         0.463513       0.500001  ...       0.471405       0.471405
min         0.000000       0.000000  ...       0.000000       0.000000
25%         0.000000       0.000000  ...       0.000000       0.000000
50%         0.000000       0.500000  ...       0.000000       0.000000
75%         1.000000       1.000000  ...       1.000000       1.000000
max         1.000000       1.000000  ...       1.000000       1.000000

[8 rows x 26 columns]
```

[6]:
```
df.drop(["Country"],axis=1,inplace=True)
df.sample(5)
```

[6]:
```
        Fever  Tiredness  Dry-Cough  ...  Contact_Dont-Know  Contact_No
Contact_Yes
95454       1          1          0  ...                  0           0
1
73645       0          0          0  ...                  0           1
0
255269      0          0          0  ...                  1           0
0
258828      0          1          0  ...                  0           0
1
248589      0          0          1  ...                  0           0
1

[5 rows x 26 columns]
```

[7]:
```
f, ax = plt.subplots(figsize=(30,30))
sns.heatmap(df.corr(), annot=True)
```

[7]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f854b5c0518>
```



[8]:
```
severity_columns = df.filter(like='Severity_').columns
```

[9]:
```
df['Severity_None'].replace({1: 'None', 0: 'No'}, inplace = True)
df['Severity_Mild'].replace({1: 'Mild', 0: 'No'}, inplace = True)
df['Severity_Moderate'].replace({1: 'Moderate', 0: 'No'}, inplace = True)
df['Severity_Severe'].replace({1: 'Severe', 0: 'No'}, inplace = True)
```

[10]:
```
df
```

[10]:
```
        Fever  Tiredness  Dry-Cough  ...  Contact_Dont-Know  Contact_No
Contact_Yes
0           1          1          1  ...                  0           0
1
```

```
1           1          1          1  ...                  0           1
0
2           1          1          1  ...                  1           0
0
3           1          1          1  ...                  0           0
1
4           1          1          1  ...                  0           1
0
...       ...        ...        ...  ...                ...         ...
...
316795      0          0          0  ...                  0           1
0
316796      0          0          0  ...                  1           0
0
316797      0          0          0  ...                  0           0
1
316798      0          0          0  ...                  0           1
0
316799      0          0          0  ...                  1           0
0

[316800 rows x 26 columns]
```

[11]:
```
df['Condition'] = df[severity_columns].values.tolist()
```

[12]:
```
def removing(list1):
    list1 = set(list1)
    list1.discard("No")
    a = ''.join(list1)
    return a
```

[13]:
```
df['Condition'] = df['Condition'].apply(removing)
```

[14]:
```
df.drop(severity_columns,axis=1,inplace=True)
```

[15]:
```
from sklearn import preprocessing
le = preprocessing.LabelEncoder()
df['Condition'] = le.fit_transform(df['Condition'])
```

[16]:
```
df['Symptoms_Score'] = df.iloc[:,:5].sum(axis=1) + df.iloc[:,6:10].sum(axis=1)
# 0-6: 1, 7-9: 1
#division = [-1, 7, 9]
#values = [0,1]
#df['Symptoms_Score'] = pd.cut(df['Symptoms_Score'], division, labels=values)
```

[17]:
```
df
```

[17]:
```
        Fever  Tiredness  Dry-Cough  ...  Contact_Yes  Condition  Symptoms_Score
0           1          1          1  ...            1          0               9
1           1          1          1  ...            0          0               9
2           1          1          1  ...            0          0               9
```

```
3          1          1          1  ...           1          1              9
4          1          1          1  ...           0          1              9
...      ...        ...        ...  ...         ...        ...            ...
316795     0          0          0  ...           0          3              0
316796     0          0          0  ...           0          3              0
316797     0          0          0  ...           1          2              0
316798     0          0          0  ...           0          2              0
316799     0          0          0  ...           0          2              0

[316800 rows x 24 columns]
```

[18]:
```
df.head(25)
```

[18]:

	Fever	Tiredness	Dry-Cough	...	Contact_Yes	Condition	Symptoms_Score
0	1	1	1	...	1	0	9
1	1	1	1	...	0	0	9
2	1	1	1	...	0	0	9
3	1	1	1	...	1	1	9
4	1	1	1	...	0	1	9
5	1	1	1	...	0	1	9
6	1	1	1	...	1	3	9
7	1	1	1	...	0	3	9
8	1	1	1	...	0	3	9
9	1	1	1	...	1	2	9
10	1	1	1	...	0	2	9
11	1	1	1	...	0	2	9
12	1	1	1	...	1	0	8
13	1	1	1	...	0	0	8
14	1	1	1	...	0	0	8
15	1	1	1	...	1	1	8
16	1	1	1	...	0	1	8
17	1	1	1	...	0	1	8
18	1	1	1	...	1	3	8
19	1	1	1	...	0	3	8
20	1	1	1	...	0	3	8
21	1	1	1	...	1	2	8
22	1	1	1	...	0	2	8
23	1	1	1	...	0	2	8
24	1	1	1	...	1	0	7

[25 rows x 24 columns]

[19]:
```
df
```

[19]:

	Fever	Tiredness	Dry-Cough	...	Contact_Yes	Condition	Symptoms_Score
0	1	1	1	...	1	0	9
1	1	1	1	...	0	0	9
2	1	1	1	...	0	0	9
3	1	1	1	...	1	1	9

```
4          1          1          1  ...           0          1          9
...      ...        ...        ...  ...         ...        ...        ...
316795     0          0          0  ...           0          3          0
316796     0          0          0  ...           0          3          0
316797     0          0          0  ...           1          2          0
316798     0          0          0  ...           0          2          0
316799     0          0          0  ...           0          2          0

[316800 rows x 24 columns]
```

[20]:
```
contact_columns = df.filter(like='Contact_').columns
```

[21]:
```
df['Contact_Dont-Know'].replace({1:'D', 0: 'No'}, inplace=True)
df['Contact_No'].replace({1:'N', 0: 'No'}, inplace=True)
df['Contact_Yes'].replace({1:'Y', 0: 'No'}, inplace=True)
```

[22]:
```
df['Contact'] = df[contact_columns].values.tolist()
```

[23]:
```
df['Contact'] = df['Contact'].apply(removing)
```

[24]:
```
df['Contact'] = le.fit_transform(df['Contact'])
```

[25]:
```
df
```

[25]:
```
        Fever  Tiredness  Dry-Cough  ...  Condition  Symptoms_Score  Contact
0           1          1          1  ...          0               9        2
1           1          1          1  ...          0               9        1
2           1          1          1  ...          0               9        0
3           1          1          1  ...          1               9        2
4           1          1          1  ...          1               9        1
...       ...        ...        ...  ...        ...             ...      ...
316795      0          0          0  ...          3               0        1
316796      0          0          0  ...          3               0        0
316797      0          0          0  ...          2               0        2
316798      0          0          0  ...          2               0        1
316799      0          0          0  ...          2               0        0

[316800 rows x 25 columns]
```

[26]:
```
data = df[['Fever', 'Tiredness', 'Dry-Cough', 'Difficulty-in-Breathing',␣
↪,'Sore-Throat', 'Nasal-Congestion', 'Runny-Nose', 'Diarrhea', 'Contact',␣
↪'Symptoms_Score']]
```

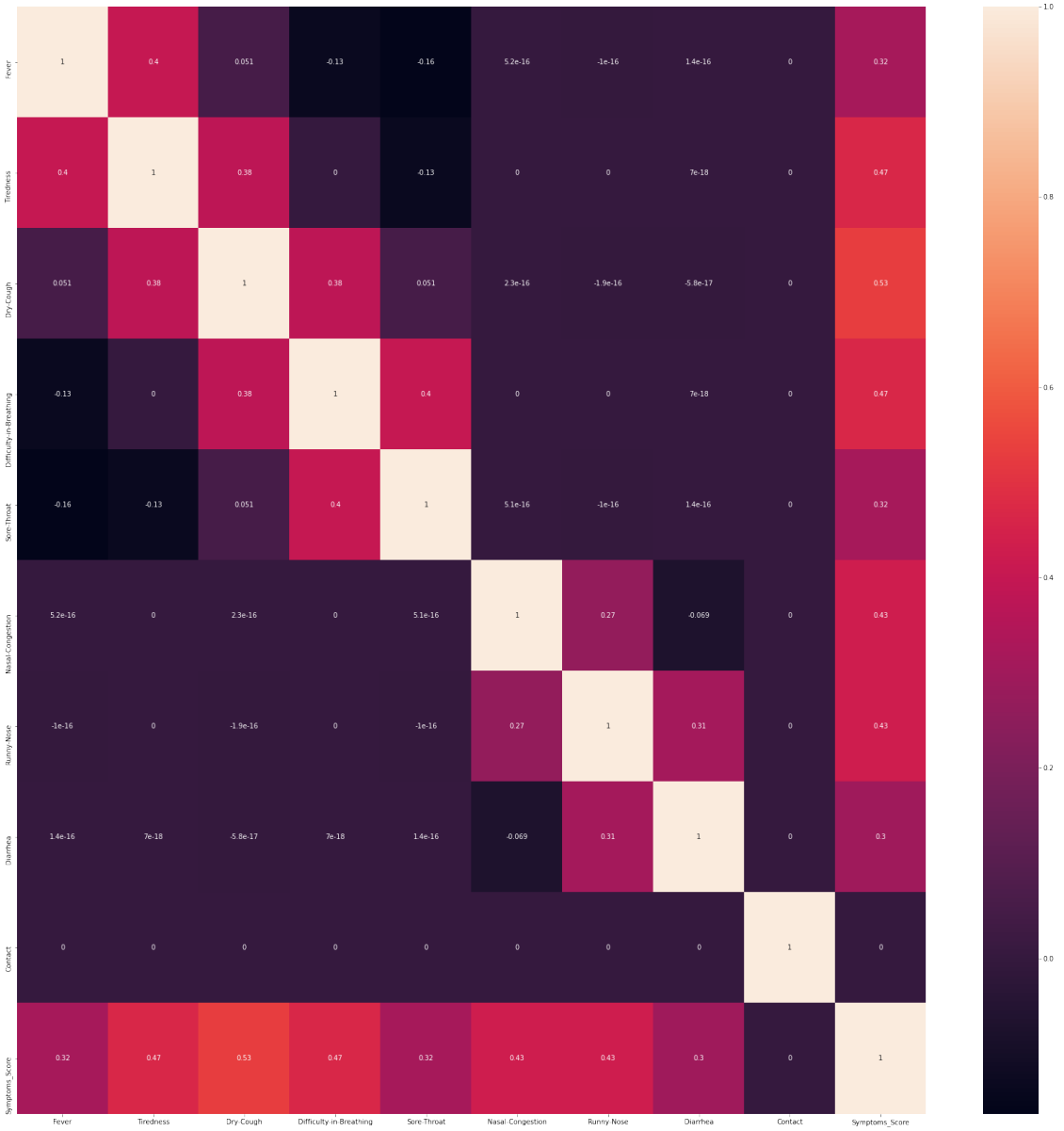
[27]:
```
data['Fever']
```

[27]:
```
0         1
1         1
2         1
3         1
4         1
        ..
316795    0
```

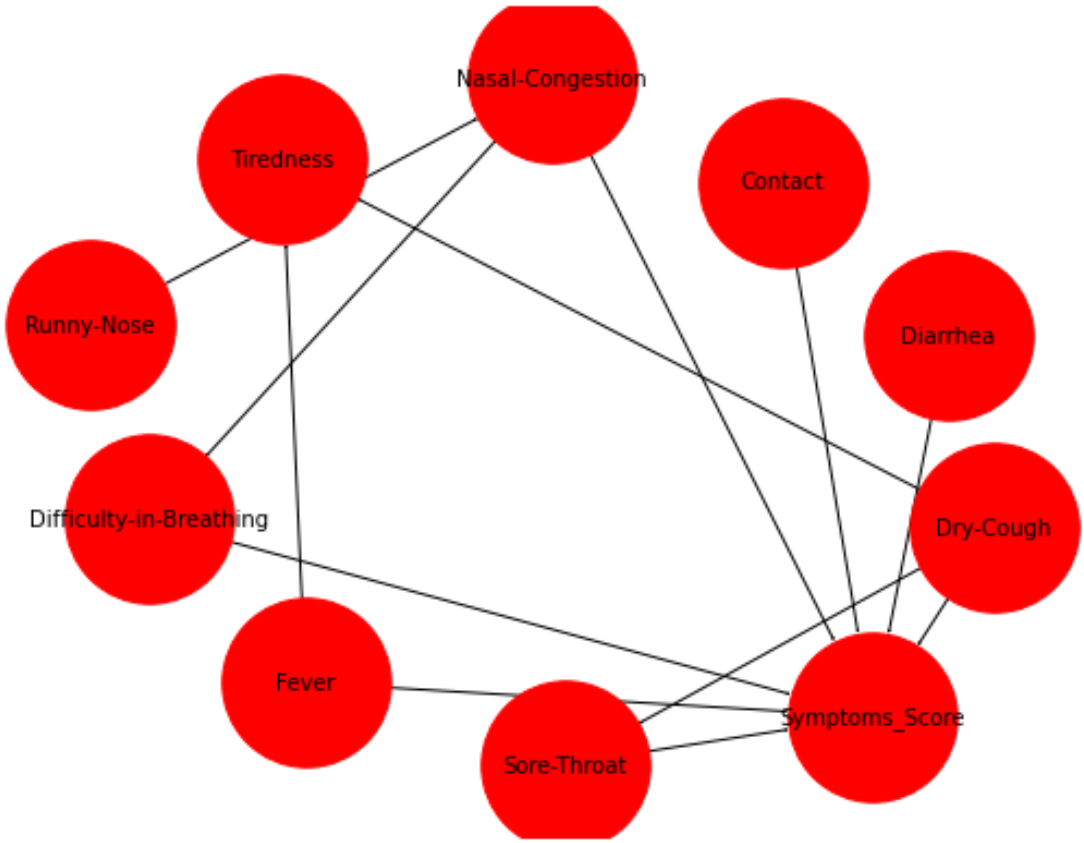
```
316796    0
316797    0
316798    0
316799    0
Name: Fever, Length: 316800, dtype: int64
```

[28]:
```
# Correlation matrix for the new dataset
f, ax = plt.subplots(figsize=(30,30))
sns.heatmap(data.corr(), annot=True)
```

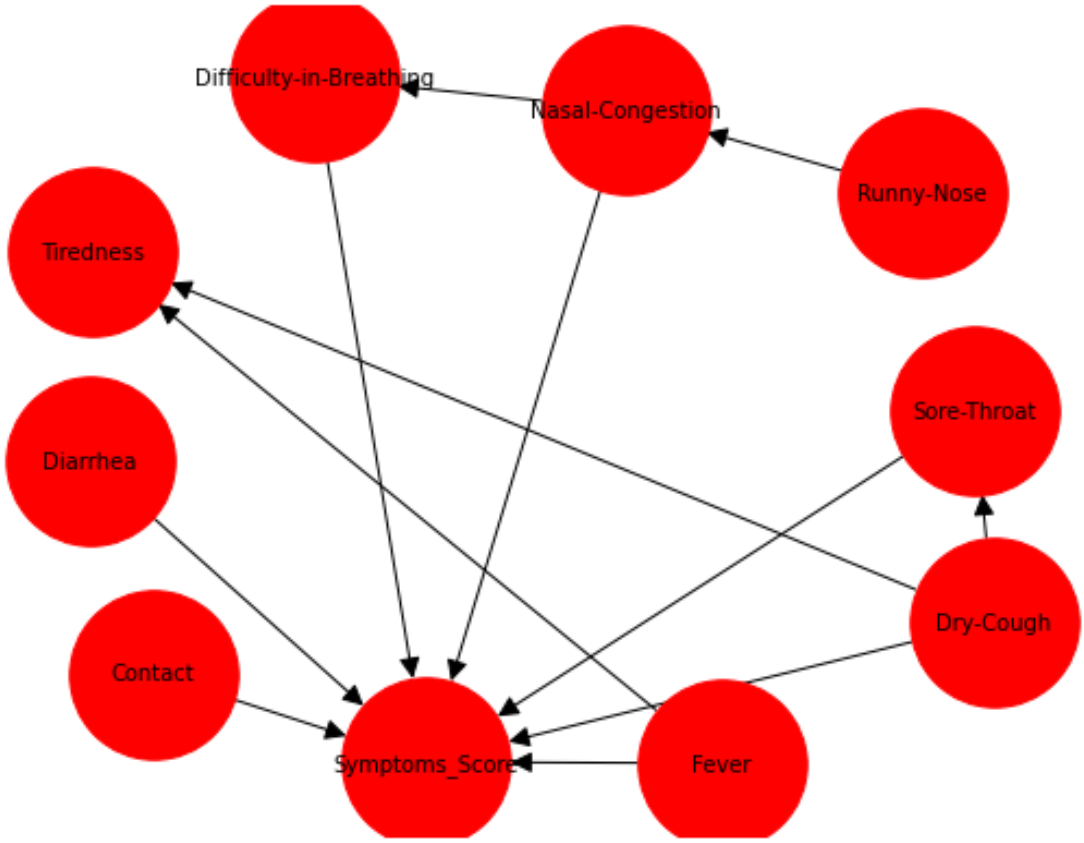
[28]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f854874df60>
```



[40]:
```
# Cognitive Casual map
casual_map = BayesianModel([('Contact', 'Symptoms_Score'), ('Fever',␣
 ↪'Symptoms_Score'),
                            ('Fever', 'Tiredness'), ('Dry-Cough',␣
 ↪'Symptoms_Score'),
                            ('Difficulty-in-Breathing', 'Symptoms_Score'),␣
 ↪('Sore-Throat','Symptoms_Score'),
                            ('Diarrhea', 'Symptoms_Score'),␣
 ↪('Nasal-Congestion', 'Symptoms_Score'),
                            ('Dry-Cough', 'Sore-Throat'), ('Dry-Cough',␣
 ↪'Tiredness'),
                            ('Runny-Nose', 'Nasal-Congestion'),␣
 ↪('Nasal-Congestion', 'Difficulty-in-Breathing')])

fig, ax = plt.subplots(figsize=(9,7))

nx.draw(covid_model, with_labels=True, node_size=6000, font_size=10,␣
 ↪arrowsize=1, node_color='red', ax=ax)
```



[41]:
```
covid_model = BayesianModel([('Contact', 'Symptoms_Score'), ('Fever',␣
↪'Symptoms_Score'),
                             ('Fever', 'Tiredness'), ('Dry-Cough',␣
↪'Symptoms_Score'),
                             ('Difficulty-in-Breathing', 'Symptoms_Score'),␣
↪('Sore-Throat','Symptoms_Score'),
                             ('Diarrhea', 'Symptoms_Score'),␣
↪('Nasal-Congestion', 'Symptoms_Score'),
                             ('Dry-Cough', 'Sore-Throat'), ('Dry-Cough',␣
↪'Tiredness'),
                             ('Runny-Nose', 'Nasal-Congestion'),␣
↪('Nasal-Congestion', 'Difficulty-in-Breathing')])

covid_model.fit(data, estimator=MaximumLikelihoodEstimator)
fig, ax = plt.subplots(figsize=(9,7))

nx.draw(covid_model, with_labels=True, node_size=6000, font_size=10,␣
↪arrowsize=20, node_color='red', ax=ax)
```



[42]:
```
covid_model.check_model()
```

[42]:
```
True
```

[43]:
```
covid_model.get_cpds()
```

[43]:
```
[<TabularCPD representing P(Contact:3) at 0x7f854561f828>,
 <TabularCPD representing P(Diarrhea:2) at 0x7f854561f080>,
 <TabularCPD representing P(Difficulty-in-Breathing:2 | Nasal-Congestion:2) at
0x7f854561f470>,
 <TabularCPD representing P(Dry-Cough:2) at 0x7f854561f6d8>,
 <TabularCPD representing P(Fever:2) at 0x7f8545619a58>,
 <TabularCPD representing P(Nasal-Congestion:2 | Runny-Nose:2) at
0x7f8545619d30>,
 <TabularCPD representing P(Runny-Nose:2) at 0x7f8545619208>,
 <TabularCPD representing P(Sore-Throat:2 | Dry-Cough:2) at 0x7f85456cc470>,
 <TabularCPD representing P(Symptoms_Score:10 | Contact:3, Diarrhea:2,
Difficulty-in-Breathing:2, Dry-Cough:2, Fever:2, Nasal-Congestion:2, Sore-
Throat:2) at 0x7f8543d4b160>,
 <TabularCPD representing P(Tiredness:2 | Dry-Cough:2, Fever:2) at
0x7f8544d498d0>]
```

[44]:
```
print(covid_model.get_cpds('Tiredness'))
```

```
+--------------+--------------+--------------+--------------+--------------+
| Dry-Cough    | Dry-Cough(0) | Dry-Cough(0) | Dry-Cough(1) | Dry-Cough(1) |
+--------------+--------------+--------------+--------------+--------------+
| Fever        | Fever(0)     | Fever(1)     | Fever(0)     | Fever(1)     |
+--------------+--------------+--------------+--------------+--------------+
| Tiredness(0) | 0.8          | 0.5          | 0.5          | 0.0          |
+--------------+--------------+--------------+--------------+--------------+
| Tiredness(1) | 0.2          | 0.5          | 0.5          | 1.0          |
+--------------+--------------+--------------+--------------+--------------+
```

[45]:
```
print(covid_model.get_cpds('Sore-Throat'))
```

```
+----------------+--------------------+--------------------+
| Dry-Cough      | Dry-Cough(0)       | Dry-Cough(1)       |
+----------------+--------------------+--------------------+
| Sore-Throat(0) | 0.7142857142857143 | 0.6666666666666666 |
+----------------+--------------------+--------------------+
| Sore-Throat(1) | 0.2857142857142857 | 0.3333333333333333 |
+----------------+--------------------+--------------------+
```

[46]:
```
print(covid_model.get_cpds('Difficulty-in-Breathing'))
```

```
+---------------------------+---------------------+---------------------+
| Nasal-Congestion          | Nasal-Congestion(0) | Nasal-Congestion(1) |
+---------------------------+---------------------+---------------------+
| Difficulty-in-Breathing(0) | 0.5                 | 0.5                 |
```

```
+---------------------------+--------------------+--------------------+
| Difficulty-in-Breathing(1) | 0.5                | 0.5                |
+---------------------------+--------------------+--------------------+
```

[47]:
```
inference = VariableElimination(covid_model)
query1 = inference.map_query(variables=['Symptoms_Score'],␣
 ↪evidence={'Dry-Cough': 1, 'Diarrhea': 1, 'Sore-Throat': 1, 'Contact': 2,␣
 ↪'Fever': 1, 'Difficulty-in-Breathing': 1, 'Nasal-Congestion': 1})
print(query1)
```

```
Finding Elimination Order: : 100%|| 2/2 [00:00<00:00, 1520.50it/s]

{'Symptoms_Score': 8}
```

[48]:
```
covid_model.get_cpds('Symptoms_Score')
```

[48]:
```
<TabularCPD representing P(Symptoms_Score:10 | Contact:3, Diarrhea:2,
Difficulty-in-Breathing:2, Dry-Cough:2, Fever:2, Nasal-Congestion:2, Sore-
Throat:2) at 0x7f8543d4b160>
```

[ ]:

[ ]: