

```
In [347]: import pandas as pd
import numpy as np
from sklearn import pipeline, preprocessing, metrics, model_selection, ensemble
from sklearn_pandas import DataFrameMapper
import seaborn as sb
```

```
In [348]: data=pd.read_excel('jugulbondi.xlsx')
data
```

Out[348]:

	age	height	weight	city	education	income	gender	body_type	complexin	drinki
0	23.0	66.0	70.0	pabna	BSC	60000.0	Female	Slim	Fair	
1	25.0	70.0	50.0	dhaka	MBA	30000.0	Female	Fatty	Lite Fair	
2	NaN	58.0	58.0	pabna	HSC	80000.0	Male	Slim	Black	
3	35.0	62.0	80.0	khulna	LLB	100000.0	Male	Medium	Lite Brown	Y
4	30.0	60.0	55.0	chittagong	MBA	40000.0	Female	Slim	Brown	
...	
245	30.0	62.0	58.0	sylhet	Honor's	0.0	Female	Slim	Lite Brown	Y
246	25.0	65.0	75.0	sylhet	BSC	20000.0	Male	Medium	Lite Brown	
247	24.0	67.0	90.0	khulna	BSC	0.0	Male	Fatty	Lite Brown	Y
248	34.0	66.0	85.0	dhaka	Honor's	55000.0	Male	Medium	Fair	
249	29.0	60.0	55.0	rangpur	Honor's	60000.0	Female	Slim	Black	Y

250 rows × 16 columns

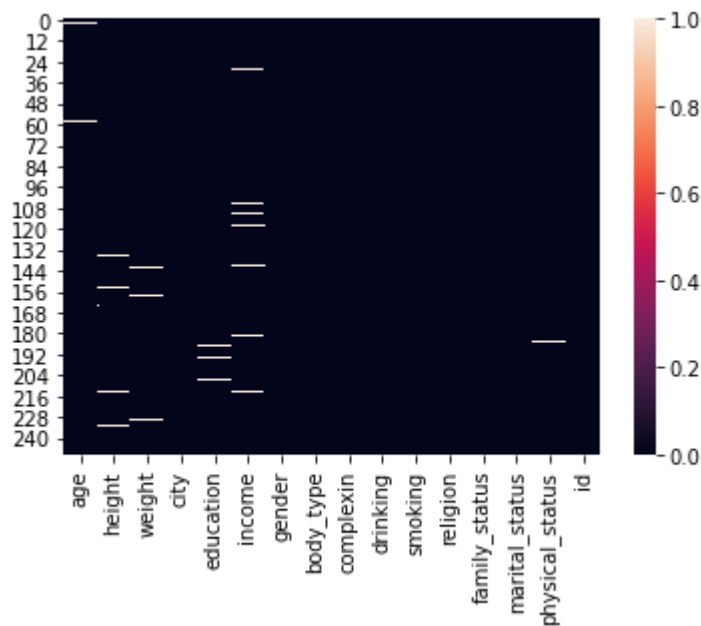


```
In [349]: data.isnull().sum()
```

```
Out[349]: age                3
height            6
weight            4
city              0
education         3
income            8
gender            0
body_type         0
complexin         0
drinking          0
smoking           0
religion          0
family_status     0
marital_status    0
physical_status   2
id                0
dtype: int64
```

```
In [350]: sb.heatmap(data.isnull())
```

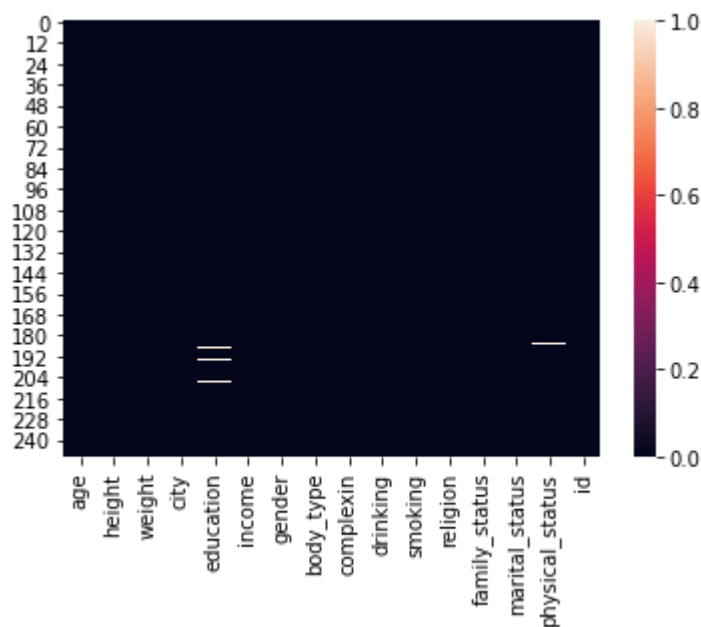
```
Out[350]: <matplotlib.axes._subplots.AxesSubplot at 0x16db815dc70>
```



```
In [351]: data["age"] = data["age"].fillna(data["age"].mode()[0])
data["height"] = data["height"].fillna(data["height"].mean())
data["weight"] = data["weight"].fillna(data["weight"].mode()[0])
data["income"] = data["income"].fillna(data["income"].mean())
```

```
In [352]: sb.heatmap(data.isnull())
```

```
Out[352]: <matplotlib.axes._subplots.AxesSubplot at 0x16db74e8760>
```



```
In [353]: data = data.dropna(axis=0)
```

```
In [354]: data
```

Out[354]:

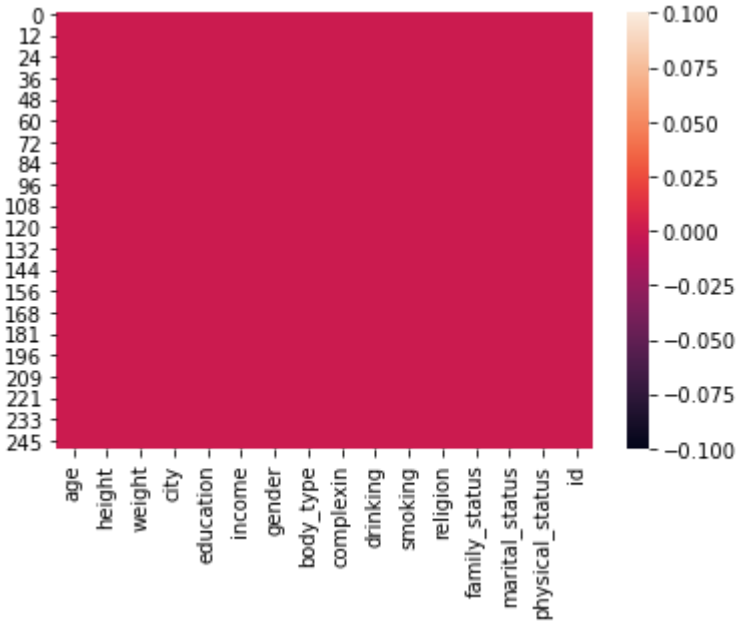
	age	height	weight	city	education	income	gender	body_type	complexin	drinki
0	23.0	66.0	70.0	pabna	BSC	60000.0	Female	Slim	Fair	
1	25.0	70.0	50.0	dhaka	MBA	30000.0	Female	Fatty	Lite Fair	
2	25.0	58.0	58.0	pabna	HSC	80000.0	Male	Slim	Black	
3	35.0	62.0	80.0	khulna	LLB	100000.0	Male	Medium	Lite Brown	Y
4	30.0	60.0	55.0	chittagong	MBA	40000.0	Female	Slim	Brown	
...
245	30.0	62.0	58.0	sylhet	Honor's	0.0	Female	Slim	Lite Brown	Y
246	25.0	65.0	75.0	sylhet	BSC	20000.0	Male	Medium	Lite Brown	
247	24.0	67.0	90.0	khulna	BSC	0.0	Male	Fatty	Lite Brown	Y
248	34.0	66.0	85.0	dhaka	Honor's	55000.0	Male	Medium	Fair	
249	29.0	60.0	55.0	rangpur	Honor's	60000.0	Female	Slim	Black	Y

245 rows × 16 columns



```
In [355]: sb.heatmap(data.isnull())
```

Out[355]: <matplotlib.axes._subplots.AxesSubplot at 0x16db75ab070>




```
In [394]: kn=pipeline_obj.score(data[X],data[Y].values.ravel())
```

```
In [395]: print("Random Forest Regressor Accuracy =",kn)
```

Random Forest Regressor Accuracy = 0.6838107404888036

```
In [396]: from sklearn.neighbors import KNeighborsClassifier
pipeline_Kn = pipeline.Pipeline([
    ('mapper',mapper),
    ("model", KNeighborsClassifier(n_neighbors=1, metric='minkowski', p=2))
])
```

```
In [397]: ngb=pipeline_Kn.fit(data[X],data[Y].values.ravel())
```

```
In [398]: ngba=pipeline_Kn.score(data[X],data[Y].values.ravel())
```

```
In [399]: print("KNeighborsClassifier Accuracy =",ngba)
```

KNeighborsClassifier Accuracy = 0.9551020408163265

```
In [400]: p=pipeline_Kn.predict(data[X])
```

```
In [401]: pipeline_rfcla = pipeline.Pipeline([
    ('mapper',mapper),
    ("model", ensemble.RandomForestClassifier(n_estimators=70))
])
```

```
In [402]: rtcla=pipeline_rfcla.fit(data[X],data[Y].values.ravel())
```

```
In [403]: rtclaa=pipeline_rfcla.score(data[X],data[Y].values.ravel())
```

```
In [404]: print("RandomForestClassifierr Accuracy =",rtclaa)
```

RandomForestClassifierr Accuracy = 0.9551020408163265

```
In [405]: from sklearn import naive_bayes
pipeline_nv = pipeline.Pipeline([
    ('mapper',mapper),
    ("model", naive_bayes.BernoulliNB())
])
```

```
In [406]: nvfit=pipeline_nv.fit(data[X],data[Y].values.ravel())
```

```
In [407]: nvacc=pipeline_nv.score(data[X],data[Y].values.ravel())
```

```
In [408]: print("Naive Bayes Accuracy =",nvacc)
```

Naive Bayes Accuracy = 0.9387755102040817

```
In [409]: temp={}
temp['age']=20
temp['height']=60
temp['weight']=70
temp['city']='dhaka'
temp['education']='LLB'
temp['income']=700000
temp['gender']='Female'
temp['body_type']='Slim'
temp['complexin']='Fair'
temp['drinking']='No'
temp['smoking']='No'
temp['religion']='Muslim'
temp['family_status']='High'
temp['marital_status']='Unmarried'
temp['physical_status']='Good'
```

```
In [410]: testData=pd.DataFrame({'i':temp}).transpose()
```

```
In [411]: testData
```

```
Out[411]:
```

	age	body_type	city	complexin	drinking	education	family_status	gender	height	income
i	20	Slim	dhaka	Fair	No	LLB	High	Female	60	700000

```
In [412]: res = ob.predict(testData)[0]
print("Using Random Forest Regressor =",res,"==",round(res))

Using Random Forest Regressor = 133.02857142857144 == 133.0
```

```
In [413]: res = ngb.predict(testData)[0]
print("Using KNeighborsClassifier =",res,"==",round(res))

Using KNeighborsClassifier = 131 == 131
```

```
In [414]: res = rtcla.predict(testData)[0]
print("Using RandomForestClassifierr =",res,"==",round(res))

Using RandomForestClassifierr = 210 == 210
```

```
In [415]: res = nvfit.predict(testData)[0]
print("Using Naive Bayes =",res,"==",round(res))

Using Naive Bayes = 165 == 165
```

```
In [ ]:
```