Final_proj_3

April 13, 2023

[1]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[67]:
```python
hosp = pd.read_csv('Hospitalisation details.csv')
```

[68]:
```python
medic = pd.read_csv('Medical Examinations.csv')
```

[4]:
```python
names = pd.read_excel('Names.xlsx')
```

[5]:
```python
hosp.head()
```

[5]:
```
  Customer ID  year month  date  children  charges Hospital tier City tier  \
0      Id2335  1992   Jul     9         0   563.84      tier - 2  tier - 3
1      Id2334  1992   Nov    30         0   570.62      tier - 2  tier - 1
2      Id2333  1993   Jun    30         0   600.00      tier - 2  tier - 1
3      Id2332  1992   Sep    13         0   604.54      tier - 3  tier - 3
4      Id2331  1998   Jul    27         0   637.26      tier - 3  tier - 3

  State ID
0    R1013
1    R1013
2    R1013
3    R1013
4    R1013
```

[6]:
```python
final_data=pd.merge(hosp, medic,how='inner', on = 'Customer ID')
```

[7]:
```python
final_data.head()
```

[7]:
```
  Customer ID  year month  date  children  charges Hospital tier City tier  \
0      Id2335  1992   Jul     9         0   563.84      tier - 2  tier - 3
1      Id2334  1992   Nov    30         0   570.62      tier - 2  tier - 1
2      Id2333  1993   Jun    30         0   600.00      tier - 2  tier - 1
3      Id2332  1992   Sep    13         0   604.54      tier - 3  tier - 3
4      Id2331  1998   Jul    27         0   637.26      tier - 3  tier - 3
```

```
  State ID    BMI  HBA1C Heart Issues Any Transplants Cancer history  \
0    R1013  17.58   4.51           No              No             No
1    R1013  17.60   4.39           No              No             No
2    R1013  16.47   6.35           No              No            Yes
3    R1013  17.70   6.28           No              No             No
4    R1013  22.34   5.57           No              No             No

  NumberOfMajorSurgeries smoker
0                      1     No
1                      1     No
2                      1     No
3                      1     No
4                      1     No
```

[8]:
```
final_data = final_data.merge(names,on='Customer ID')
```

[9]:
```
final_data.shape
```

[9]:
```
(2335, 17)
```

[10]:
```
final_data.columns = final_data.columns.str.lower()
final_data.columns = final_data.columns.str.replace(' ', '_')
final_data.columns
```

[10]:
```
Index(['customer_id', 'year', 'month', 'date', 'children', 'charges',
       'hospital_tier', 'city_tier', 'state_id', 'bmi', 'hba1c',
       'heart_issues', 'any_transplants', 'cancer_history',
       'numberofmajorsurgeries', 'smoker', 'name'],
      dtype='object')
```

[11]:
```
(final_data=='?').sum()
```

[11]:
```
customer_id               0
year                      2
month                     3
date                      0
children                  0
charges                   0
hospital_tier             1
city_tier                 1
state_id                  2
bmi                       0
hba1c                     0
heart_issues              0
any_transplants           0
cancer_history            0
```

```
numberofmajorsurgeries    0
smoker                    2
name                      0
dtype: int64
```

[12]:
```
miss_perc = (final_data =='?').sum(axis = 1)/final_data.shape[1]
```

[13]:
```
miss_perc[miss_perc > 0]
```

[13]:
```
11      0.058824
13      0.058824
17      0.117647
542     0.058824
1046    0.058824
1049    0.058824
1700    0.058824
1775    0.058824
2165    0.058824
2332    0.058824
dtype: float64
```

[14]:
```
miss_perc_colm = (final_data =='?').sum(axis = 0)/final_data.shape[0]
```

[15]:
```
miss_perc_colm.sort_values(ascending=False)
```

[15]:
```
month                     0.001285
state_id                  0.000857
year                      0.000857
smoker                    0.000857
city_tier                 0.000428
hospital_tier             0.000428
date                      0.000000
children                  0.000000
charges                   0.000000
name                      0.000000
bmi                       0.000000
hba1c                     0.000000
heart_issues              0.000000
any_transplants           0.000000
cancer_history            0.000000
numberofmajorsurgeries    0.000000
customer_id               0.000000
dtype: float64
```

[16]:
```
final_noq = final_data.drop(index = miss_perc[miss_perc>0].index)
```

[17]:
```
final_noq.shape
```

```
[17]: (2325, 17)
```

```
[18]: final_noq[['city_tier', 'hospital_tier']]
```

```
[18]:      city_tier hospital_tier
      0     tier - 3      tier - 2
      1     tier - 1      tier - 2
      2     tier - 1      tier - 2
      3     tier - 3      tier - 3
      4     tier - 3      tier - 3
      …        …             …
      2329  tier - 3      tier - 1
      2330  tier - 2      tier - 1
      2331  tier - 3      tier - 1
      2333  tier - 3      tier - 2
      2334  tier - 3      tier - 1

      [2325 rows x 2 columns]
```

```
[19]: from sklearn.preprocessing import OrdinalEncoder
      ordinal = OrdinalEncoder(categories=[['tier - 3','tier - 2','tier - 1'],['tier␣
       ↪- 3', 'tier - 2', 'tier - 1']])
      final_noq[['city_tier_ord','hospital_tier_ord']] = ordinal.
       ↪fit_transform(final_noq[['city_tier','hospital_tier']])
```

```
[20]: pd.crosstab(final_noq['city_tier_ord'],final_noq['city_tier'])
```

```
[20]: city_tier      tier - 1  tier - 2  tier - 3
      city_tier_ord
      0.0                   0         0       789
      1.0                   0       807         0
      2.0                 729         0         0
```

```
[21]: pd.crosstab(final_noq['hospital_tier_ord'],final_noq['hospital_tier'])
```

```
[21]: hospital_tier      tier - 1  tier - 2  tier - 3
      hospital_tier_ord
      0.0                       0         0       691
      1.0                       0      1334         0
      2.0                     300         0         0
```

```
[22]: final_noq.head()
```

```
[22]:   customer_id  year month  date  children  charges hospital_tier city_tier  \
      0      Id2335  1992   Jul     9         0   563.84      tier - 2  tier - 3
      1      Id2334  1992   Nov    30         0   570.62      tier - 2  tier - 1
      2      Id2333  1993   Jun    30         0   600.00      tier - 2  tier - 1
```

```
3      Id2332  1992   Sep    13          0   604.54      tier - 3  tier - 3
4      Id2331  1998   Jul    27          0   637.26      tier - 3  tier - 3

  state_id    bmi  hba1c heart_issues any_transplants cancer_history  \
0    R1013  17.58   4.51           No              No             No
1    R1013  17.60   4.39           No              No             No
2    R1013  16.47   6.35           No              No            Yes
3    R1013  17.70   6.28           No              No             No
4    R1013  22.34   5.57           No              No             No

  numberofmajorsurgeries smoker                             name  \
0                      1     No                German, Mr.  Aaron K
1                      1     No              Rosendahl, Mr.  Evan P
2                      1     No                 Albano, Ms.  Julie
3                      1     No  Riveros Gonzalez, Mr.  Juan D. Sr.
4                      1     No               Brietzke, Mr.  Jordan

   city_tier_ord  hospital_tier_ord
0            0.0                1.0
1            2.0                1.0
2            2.0                1.0
3            0.0                0.0
4            0.0                0.0
```

[ ]:

[ ]:

[ ]:

[ ]:

[23]:
```
final_noq.head()
```

[23]:
```
  customer_id  year month  date  children  charges hospital_tier city_tier  \
0      Id2335  1992   Jul     9         0   563.84      tier - 2  tier - 3
1      Id2334  1992   Nov    30         0   570.62      tier - 2  tier - 1
2      Id2333  1993   Jun    30         0   600.00      tier - 2  tier - 1
3      Id2332  1992   Sep    13         0   604.54      tier - 3  tier - 3
4      Id2331  1998   Jul    27         0   637.26      tier - 3  tier - 3

  state_id    bmi  hba1c heart_issues any_transplants cancer_history  \
0    R1013  17.58   4.51           No              No             No
1    R1013  17.60   4.39           No              No             No
2    R1013  16.47   6.35           No              No            Yes
3    R1013  17.70   6.28           No              No             No
4    R1013  22.34   5.57           No              No             No
```

```
  numberofmajorsurgeries smoker                              name  \
0                      1     No               German, Mr.  Aaron K
1                      1     No             Rosendahl, Mr.  Evan P
2                      1     No                 Albano, Ms.  Julie
3                      1     No  Riveros Gonzalez, Mr.  Juan D. Sr.
4                      1     No              Brietzke, Mr.  Jordan

   city_tier_ord  hospital_tier_ord
0            0.0                1.0
1            2.0                1.0
2            2.0                1.0
3            0.0                0.0
4            0.0                0.0
```

[24]:
```
vc = final_noq.state_id.value_counts()   # frequency of each category
vc[:3].index
```

[24]:
```
Index(['R1013', 'R1011', 'R1012'], dtype='object')
```

[25]:
```
for i in vc[:3].index:
    var_name = 'state_id_' +i
    print(var_name)
    final_noq[var_name] = 0
    final_noq.loc[final_noq.state_id == i,var_name] = 1
```

```
state_id_R1013
state_id_R1011
state_id_R1012
```

[26]:
```
final_noq.state_id.value_counts()
```

[26]:
```
R1013    609
R1011    574
R1012    572
R1024    159
R1026     84
R1021     70
R1016     64
R1025     40
R1023     38
R1017     36
R1019     26
R1022     14
R1014     13
R1015     11
R1018      9
```

```
R1020      6
Name: state_id, dtype: int64
```

[27]:
```
final_noq['state_id_R1013'].value_counts()
```

[27]:
```
0    1716
1     609
Name: state_id_R1013, dtype: int64
```

[28]:
```
final_noq.numberofmajorsurgeries.unique()
```

[28]:
```
array(['1', 'No major surgery', '2', '3'], dtype=object)
```

[29]:
```
final_noq.loc[final_noq.numberofmajorsurgeries == 'No major␣
 ↪surgery','numberofmajorsurgeries' ] = 0
```

[30]:
```
final_noq.numberofmajorsurgeries = final_noq.numberofmajorsurgeries.astype(int)
```

[31]:
```
final_noq.year = final_noq.year.astype(int)
```

[32]:
```
final_noq['age'] = 2023- final_noq.year
```

[33]:
```
final_noq['title'] = final_noq.name.str.split('[,.]').str[1].str.strip()
```

[34]:
```
final_noq.title.value_counts()
```

[34]:
```
Mr     1160
Ms     1023
Mrs     142
Name: title, dtype: int64
```

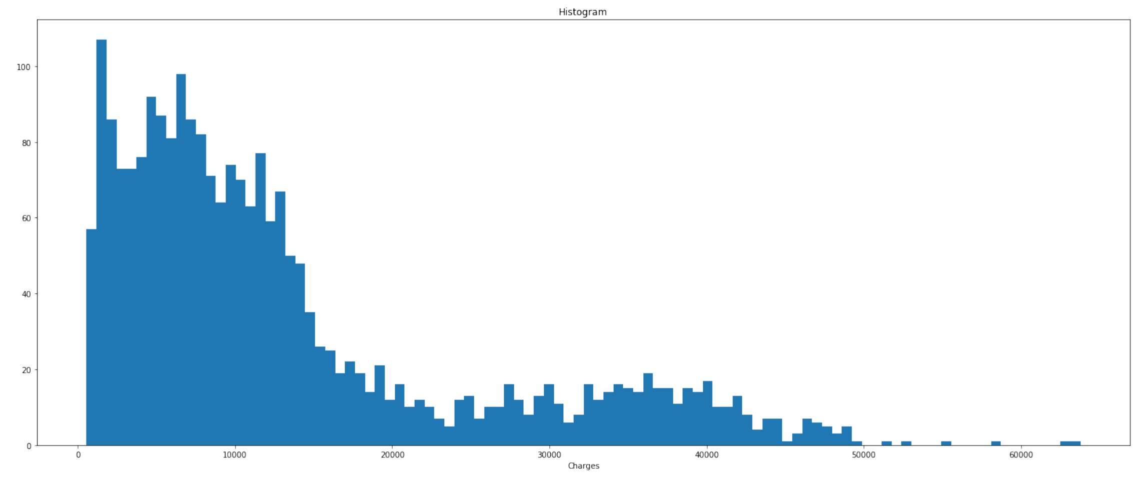
[35]:
```
final_noq['gender'] = 'female'
final_noq.loc[final_noq.title == 'Mr', 'gender' ] = 'male'
```

[36]:
```
final_noq.loc[final_noq.title == 'Mrs']
```

[36]:
```
     customer_id  year month  date  children   charges hospital_tier  \
24        Id2311  2001   Aug    19         0    964.71      tier - 3
172       Id2163  2004   Dec    27         0   1863.45      tier - 3
197       Id2138  2004   Jun    12         0   2094.10      tier - 3
328       Id2007  1993   Sep    25         0   3162.02      tier - 2
348       Id1987  2003   Dec     5         0   3300.70      tier - 2
…            …  …     …   …      …         …             …
1790       Id545  1963   Jul     4         0  18208.34      tier - 1
1808       Id527  1963   Dec     6         0  18883.33      tier - 1
1811       Id524  1963   Oct    20         0  18954.56      tier - 1
1839       Id496  1966   Aug    10         0  19995.29      tier - 1
```

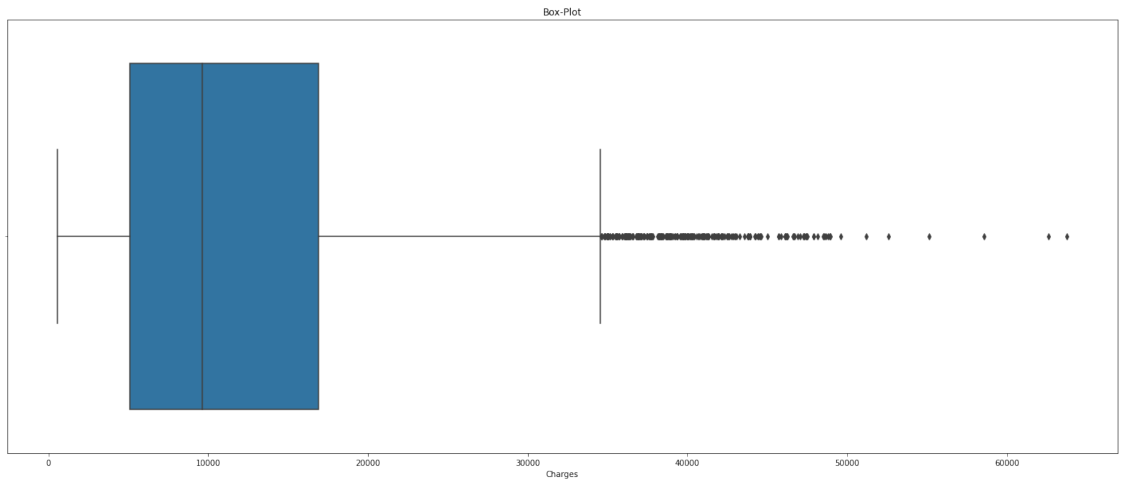
```
1848       Id487  1962   Jul     2         0  20354.50      tier - 3

     city_tier state_id    bmi  …  smoker                     name  \
24    tier - 2    R1013  25.19  …      No        Keys, Mrs.  Kathleen
172   tier - 1    R1025  27.06  …      No    Stanislav, Mrs.  Grace H
197   tier - 2    R1025  27.74  …      No        Padula, Mrs.  Lauren
328   tier - 3    R1013  25.61  …      No      Martin, Mrs.  Kristen M
348   tier - 2    R1025  30.54  …      No  Mendez-Karr, Mrs.  Cynthia
…          …        …      …  …     …                           …
1790  tier - 2    R1026  44.20  …      No      Shigezumi, Mrs.  Teiko
1808  tier - 1    R1026  46.19  …      No      Hughey, Mrs.  Ashley E
1811  tier - 1    R1026  46.40  …      No      Rogers, Mrs.  Anita L.
1839  tier - 3    R1026  51.74  …      No       Oehlke, Mrs.  Jessica
1848  tier - 2    R1026  49.77  …      No        Argall, Mrs.  Tara R

     city_tier_ord hospital_tier_ord  state_id_R1013 state_id_R1011  \
24             1.0               0.0               1              0
172            2.0               0.0               0              0
197            1.0               0.0               0              0
328            0.0               1.0               1              0
348            1.0               1.0               0              0
…              …                 …               …              …
1790           1.0               2.0               0              0
1808           2.0               2.0               0              0
1811           2.0               2.0               0              0
1839           0.0               2.0               0              0
1848           1.0               0.0               0              0

     state_id_R1012  age  title  gender
24                0   22    Mrs  female
172               0   19    Mrs  female
197               0   19    Mrs  female
328               0   30    Mrs  female
348               0   20    Mrs  female
…              …  …    …      …
1790              0   60    Mrs  female
1808              0   60    Mrs  female
1811              0   60    Mrs  female
1839              0   57    Mrs  female
1848              0   61    Mrs  female

[142 rows x 25 columns]
```

[37]:
```
plt.figure(figsize = (25,10))
plt.hist(final_noq.charges, bins = 100)
plt.xlabel('Charges')
plt.title('Histogram')
plt.show()
```

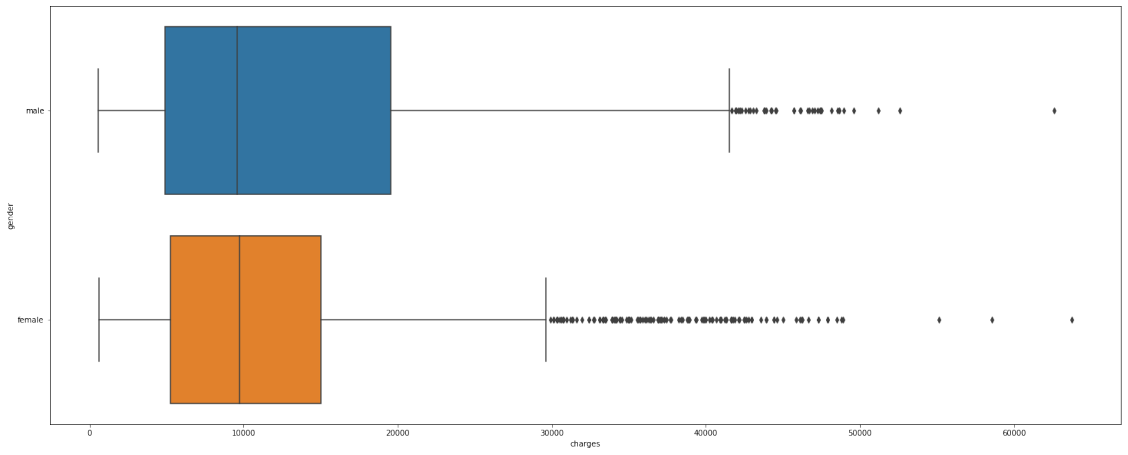


[38]:
```
plt.figure(figsize = (25,10))
sns.boxplot(final_noq.charges)
plt.xlabel('Charges')
plt.title('Box-Plot')
plt.show()
```

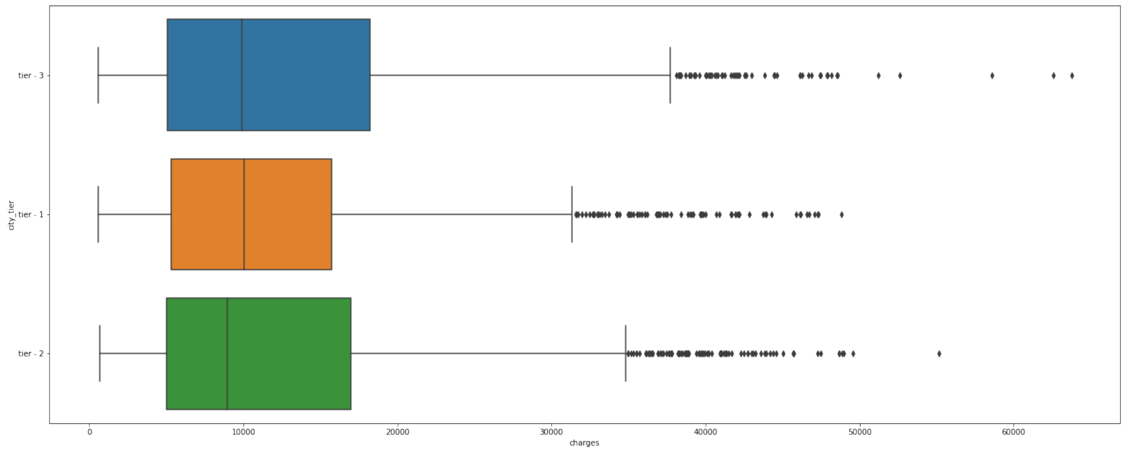
```
/usr/local/lib/python3.7/site-packages/seaborn/_decorators.py:43: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only
valid positional argument will be `data`, and passing other arguments without an
explicit keyword will result in an error or misinterpretation.
  FutureWarning
```



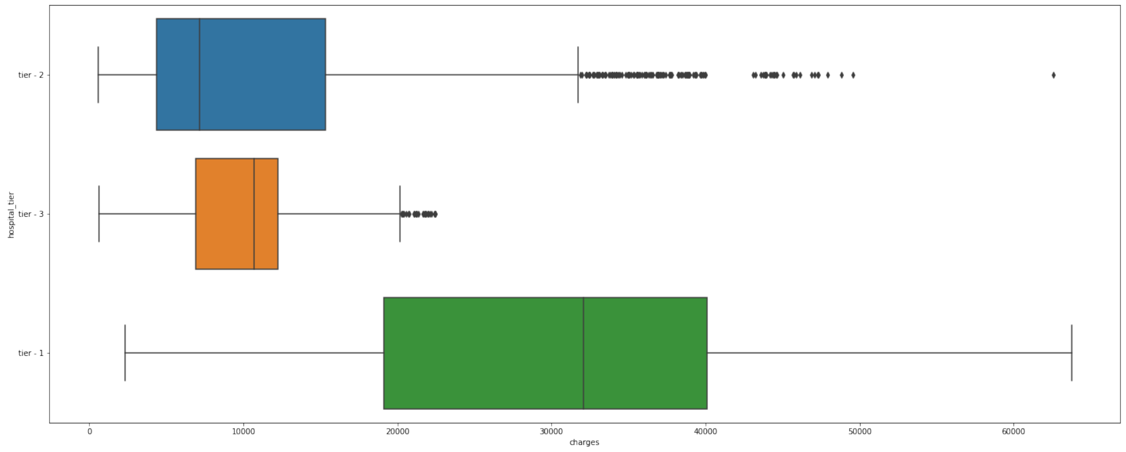
[39]:
```
plt.figure(figsize =(25,10))
sns.boxplot( x='charges',y='gender', data= final_noq)
plt.show()
```



[40]:
```
plt.figure(figsize =(25,10))
sns.boxplot( x='charges',y='city_tier', data= final_noq)
plt.show()
```

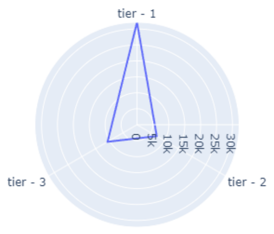


[41]:
```
plt.figure(figsize =(25,10))
sns.boxplot( x='charges',y='hospital_tier', data= final_noq)
plt.show()
```



[42]:
```
median = final_noq.groupby('hospital_tier')[['charges']].median().reset_index()
```

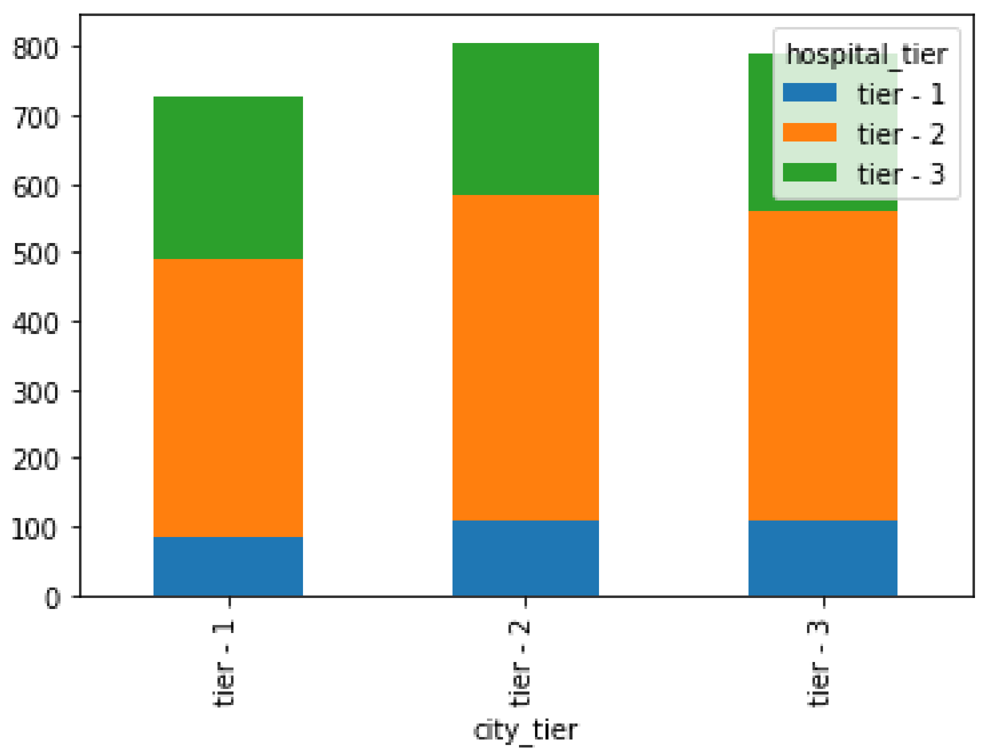
[43]:
```
import plotly.express as px
fig = px.line_polar(median, r='charges', theta='hospital_tier', line_close=True)
fig.show()
```



[44]:
```
pd.crosstab(final_noq.city_tier, final_noq.hospital_tier)
```

[44]:
```
hospital_tier  tier - 1  tier - 2  tier - 3
city_tier
tier - 1             85       403       241
tier - 2            106       479       222
tier - 3            109       452       228
```

[45]:
```
pd.crosstab(final_noq.city_tier, final_noq.hospital_tier).plot.bar(stacked =␣
 ↪True)
plt.show()
```



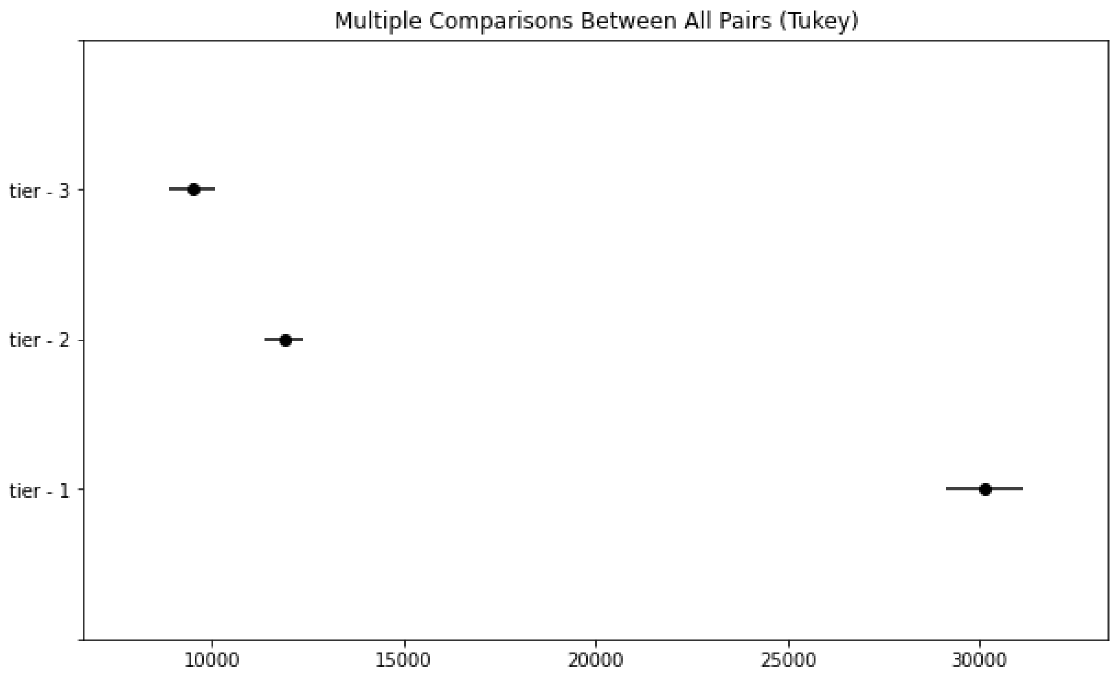
```
[46]: from statsmodels.formula.api import ols
      import statsmodels.api  as sm

      mod = ols('charges ~ hospital_tier', data = final_noq).fit()
      res = sm.stats.anova_lm(mod)
```

```
[47]: from statsmodels.stats.multicomp import pairwise_tukeyhsd

      res_tukey = pairwise_tukeyhsd(final_noq.charges, final_noq.hospital_tier)
      res_tukey.summary()
```

```
[47]: <class 'statsmodels.iolib.table.SimpleTable'>
```

```
[48]: res_tukey.plot_simultaneous()
      plt.show()
```

```
/usr/local/lib/python3.7/site-
packages/statsmodels/sandbox/stats/multicomp.py:775: UserWarning:

FixedFormatter should only be used together with FixedLocator
```



```
[49]: import scipy.stats as stats

      sample1 = final_noq.loc[final_noq.smoker == 'yes', 'charges']
      sample2 = final_noq.loc[final_noq.smoker != 'yes', 'charges']
      stats.ttest_ind(sample1, sample2)
```

```
[49]: Ttest_indResult(statistic=74.15560699695726, pvalue=0.0)
```

```
[50]: mod = ols('charges ~ city_tier', data = final_noq).fit()
      res = sm.stats.anova_lm(mod)
      res
```

```
[50]:                 df        sum_sq       mean_sq         F    PR(>F)
      city_tier      2.0  4.092192e+08  2.046096e+08  1.454356  0.233763
      Residual    2322.0  3.266763e+11  1.406874e+08       NaN       NaN
```

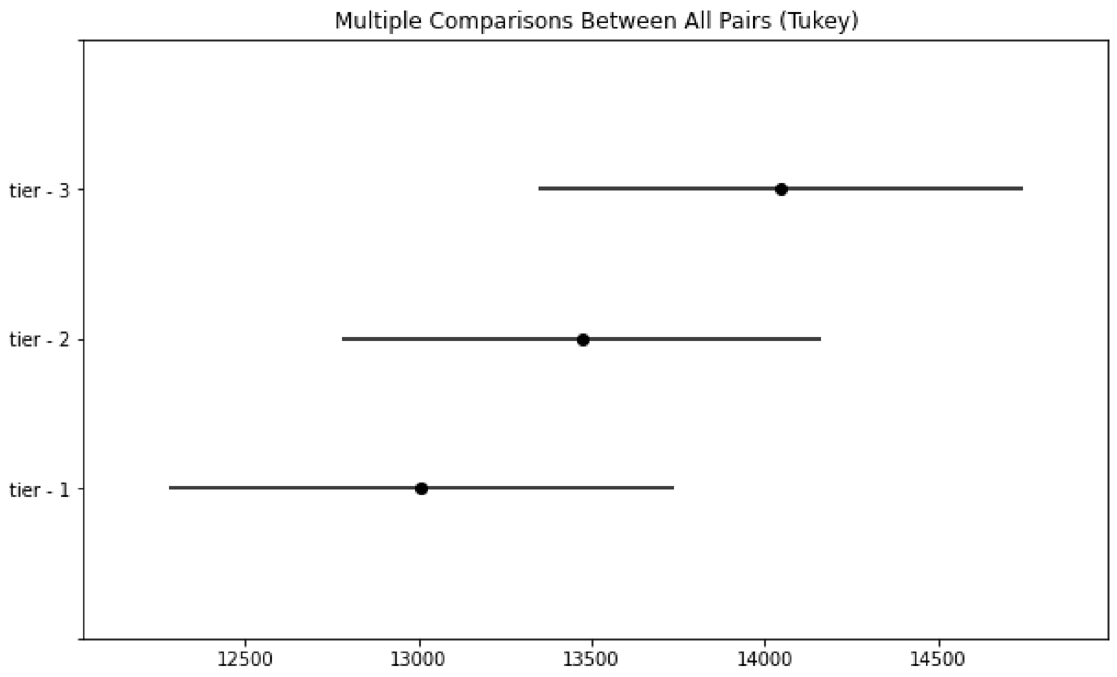
```
[51]: res_tukey = pairwise_tukeyhsd(final_noq.charges, final_noq.city_tier)
      res_tukey.summary()
```

```
[51]: <class 'statsmodels.iolib.table.SimpleTable'>
```

```
[52]: res_tukey.plot_simultaneous()
      plt.show()
```

```
/usr/local/lib/python3.7/site-
```

```
packages/statsmodels/sandbox/stats/multicomp.py:775: UserWarning:

FixedFormatter should only be used together with FixedLocator
```



[53]:
```
observed_table = pd.crosstab(final_noq.smoker, final_data.heart_issues)
```

[54]:
```
chi, p, df, expected = stats.chi2_contingency(observed_table)
```

[55]:
```
chi, p, df, expected
```

[55]:
```
(0.08588150449910657,
 0.7694797581780767,
 1,
 array([[1111.30967742,  727.69032258],
        [ 293.69032258,  192.30967742]]))
```

[56]:
```
data = final_noq.drop(columns = ['name', 'year', 'month',␣
 ↪'date','hospital_tier',
               'city_tier', 'state_id' , 'title'])
```

[57]:
```
corr_plot = data.select_dtypes(exclude='object').corr()
ma = np.ones_like(corr_plot)
ma[np.tril_indices_from(ma)] = 0
```

[58]:
```
data_2 = pd.get_dummies(data, drop_first=True)
data_2.reset_index(drop=True, inplace = True)
```

[59]:
```
model_data = data_2.drop(columns = 'charges')
model_data.head()
model_data['charges'] = data_2.charges
model_data.head()
```

[59]:
```
   children    bmi  hba1c  numberofmajorsurgeries  city_tier_ord  \
0         0  17.58   4.51                       1            0.0
1         0  17.60   4.39                       1            2.0
2         0  16.47   6.35                       1            2.0
3         0  17.70   6.28                       1            0.0
4         0  22.34   5.57                       1            0.0

   hospital_tier_ord  state_id_R1013  state_id_R1011  state_id_R1012  age  \
0                1.0               1               0               0   31
1                1.0               1               0               0   31
2                1.0               1               0               0   30
3                0.0               1               0               0   31
4                0.0               1               0               0   25

   …  customer_id_Id996  customer_id_Id997  customer_id_Id998  \
0  …                  0                  0                  0
1  …                  0                  0                  0
2  …                  0                  0                  0
3  …                  0                  0                  0
4  …                  0                  0                  0

   customer_id_Id999  heart_issues_yes  any_transplants_yes  \
0                  0                 0                    0
1                  0                 0                    0
2                  0                 0                    0
3                  0                 0                    0
4                  0                 0                    0

   cancer_history_Yes  smoker_yes  gender_male  charges
0                   0           0            1   563.84
1                   0           0            1   570.62
2                   1           0            0   600.00
3                   0           0            1   604.54
4                   0           0            1   637.26

[5 rows x 2340 columns]
```

[60]:
```
model_data.columns = model_data.columns.str.lower()
```

[61]:
```
# converting y to categorical for stratified k fold
y_class = pd.cut(model_data.charges, bins = 4, labels= [1,2,3,4])
X = model_data.drop(columns = 'charges')
```

[62]:
```
from sklearn.model_selection import KFold, StratifiedKFold, RandomizedSearchCV,␣
↪train_test_split
from sklearn.ensemble import RandomForestRegressor

from sklearn.linear_model import SGDRegressor
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error as mse, r2_score
from sklearn.preprocessing import StandardScaler
import xgboost as xgb
from sklearn.pipeline import Pipeline
from xgboost import XGBRegressor
folds = StratifiedKFold(n_splits=5, shuffle = True, random_state=12)
sgd_rmse_train = {}
sgd_rmse_test = {}
i= 1
```

[63]:
```
for train_index, test_index in folds.split(X,y_class):

    train, test = model_data.loc[train_index,], model_data.loc[test_index,]

    # standardization :
    sc = StandardScaler()
    sc.fit(train)
    train_std = sc.transform(train)
    test_std = sc.transform(test)


    x_train , x_test = train_std[:, :-1], test_std[:, :-1]
    y_train, y_test = train_std[:, -1], test_std[:,-1]
    sgd = SGDRegressor(max_iter=100)
```

[64]:
```
    space = dict()
    space['penalty'] = ['l1', 'l2', 'elasticnet']
    space['l1_ratio'] = [0,.1,.2,.8,1]
    space['alpha'] = [1e-5, 1e-4, 1e-3, 1e-2, 1e-1, 1, 10, 100, 1000,10000]
    space['learning_rate'] = ['constant', 'adaptive']
    space['eta0']=[1e-5, 1e-4, 1e-3, 1e-2, 1e-1 , 2e-1, 3e-1, 5e-1, 8e-1, 4e-1,␣
↪8e-1, 1, 10, 100]
```

[65]:
```
search = RandomizedSearchCV(sgd, space,
                                cv=5, refit=True, return_train_score = True,
                                random_state = 12)
```

[ ]:

```
```

[66]:

```
earch = RandomizedSearchCV(xgb, space, scoring='neg_root_mean_squared_error',
                                 cv=fold_inner, refit=True, return_train_score␣
 ↪= True,
                                 random_state = 12)
```

```
      ␣
↪---------------------------------------------------------------------------

       NameError                                 Traceback (most recent call␣
↪last)

       <ipython-input-66-8055994c2f9c> in <module>
         1 earch = RandomizedSearchCV(xgb, space,␣
↪scoring='neg_root_mean_squared_error',
   ----> 2                                  cv=fold_inner, refit=True,␣
↪return_train_score = True,
         3                                  random_state = 12)


       NameError: name 'fold_inner' is not defined
```

[ ]:

```
result = search.fit(x_train, y_train)

train_pred = result.predict(x_train)
test_pred  = result.predict(x_test)
```

[ ]:

```
sgd_rmse_train.update({'Fold{}'.format(i): round(mse(y_true = y_train, y_pred =␣
 ↪train_pred, squared = False),3)})
sgd_rmse_test.update({'Fold{}'.format(i): round(mse(y_true = y_test, y_pred =␣
 ↪test_pred, squared = False),3)})
i += 1
```

[ ]:

```
train, test = train_test_split(model_data, test_size = 0.25, random_state = 12)

# standardization :
sc = StandardScaler()
sc.fit(train)
trn_std = sc.transform(train)
tst_std = sc.transform(test)
```

```
# getting X and y :
x_train, x_test = trn_std[:, :-1], tst_std[:,:-1]
y_train, y_test = trn_std[:, -1], tst_std[:,-1]

# sgd regression with hyperparameter tuning :
rf = RandomForestRegressor(random_state = 12)

# define search space
space = dict()
space['n_estimators'] = [10, 100, 500]
space['max_features'] = [2, 3, 4, 5, 6]

# define search
search = RandomizedSearchCV(rf, space, scoring='neg_root_mean_squared_error',
                            cv=5, refit=True, return_train_score = True,
                                random_state = 12, n_iter = 3
                           )
# execute search
result = search.fit(x_train,y_train)
```

```
[ ]: importance = pd.Series(result.best_estimator_.feature_importances_, index =␣
      ↪train.drop(columns= 'charges').columns)
```

```
[ ]: impvars = importance.sort_values(ascending=False)[:6]
```

```
[ ]: pred_data = pd.DataFrame({'Name' : ['Christopher, Ms. Jayna'],
                             'DOB' : ['12/28/1988'],
                             'city_tier' : ['tier - 1'], 'children' :[ 2],
                              'HbA1c' : [5.8],
                              'smoker_yes' : [1],
                              'heart_issues_yes' : [0],
                              'any_transplants_yes' : [0],
                              'numberofmajorsurgeries' :[ 0],
                              'cancer_history_yes' : [1],
                              'hospital_tier' : ['tier - 1'],
                              'bmi' : [85/(1.70 **2)],
                              'state_id_R1011' : [1]
                             })
```

```
[ ]: pred_data
```

```
[ ]:
```