# Predicting Customer Costs - Multiple Linear Regression

## Patrick Herlihy

## 12/14/22

Research Question: Which variables are most important to predicting customers charges. Can these variables be used to predict additional charge to customers?

The goal of this data analysis is to better understand what factors are driving up additional costs for customers. Using this information hospitals can better predict customer costs upfront prior to admitting them.

# Data Cleaning/Exploration

Steps:

1. **Import the data from csv to dataframe.**
2. **Check for null values and duplicates.**
3. **Relabel the improperly labeled columns.**
4. **Review the summary of statistics.**
5. **Explore the dataset and generate visualizations using plotnine.**
6. **Export the prepared data to csv.**

In [1]:
```
import pandas as pd
from statsmodels.formula.api import ols
from statsmodels.api import qqplot
from itertools import product
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotnine as p9
```

In [2]:
```
#Reads CSV to data frame, sets case order to index
med_predict= pd.read_csv('/Users/herlihpj/Desktop/Data Analytics/D208 - Predictive Modeling/Task 1/Medical/medical_clean
                         index_col=0)
```

In [3]:
```
#Check for Null
print('Summary of Null: ')
print(med_predict.isna().sum())
#Check for duplicated data
duplicates=med_predict.duplicated()
print('Duplicates: ', duplicates.sum())
```

```
Summary of Null:
Customer_id           0
Interaction           0
UID                   0
City                  0
State                 0
County                0
Zip                   0
Lat                   0
Lng                   0
Population            0
Area                  0
TimeZone              0
Job                   0
Children              0
Age                   0
Income                0
Marital               0
Gender                0
ReAdmis               0
VitD_levels           0
Doc_visits            0
Full_meals_eaten      0
vitD_supp             0
Soft_drink            0
Initial_admin         0
HighBlood             0
Stroke                0
Complication_risk     0
Overweight            0
Arthritis             0
Diabetes              0
Hyperlipidemia        0
BackPain              0
Anxiety               0
Allergic_rhinitis     0
Reflux_esophagitis    0
Asthma                0
Services              0
Initial_days          0
TotalCharge           0
Additional_charges    0
Item1                 0
Item2                 0
Item3                 0
```

```
Item4                0
Item5                0
Item6                0
Item7                0
Item8                0
dtype: int64
Duplicates:  0
```

In [4]:
```
#Add column titles to last 8 columns of the data frame
med_predict.rename(columns = {'Item1':'Timely admission',
    'Item2':'Timely treatment',
    'Item3':'Timely visits',
    'Item4':'Reliability',
    'Item5':'Options',
    'Item6':'Hours of treatment',
    'Item7':'Courteous staff',
    'Item8':'Evidence of active listening'},
inplace=True)
```

In [5]:
```
# Dropping columns that I think are not relavant to the analysis
med_predict = med_predict.drop(columns= ["Customer_id", "Interaction", "UID", "City", "County", "Zip", "Lat",
                            "Lng",'Job', "Population" ])
```

In [6]:
```
#Explore the data/statistics
print(med_predict.head())
print(med_predict.describe())
print(med_predict.info())
```

```
          State      Area          TimeZone  Children  Age    Income  \
CaseOrder
1            AL  Suburban   America/Chicago         1   53  86575.93
2            FL     Urban   America/Chicago         3   51  46805.99
3            SD  Suburban   America/Chicago         3   53  14370.14
4            MN  Suburban   America/Chicago         0   78  39741.49
5            VA     Rural  America/New_York         1   22   1209.56

            Marital  Gender ReAdmis  VitD_levels  ...  TotalCharge  \
CaseOrder                                        ...
1          Divorced    Male      No    19.141466  ...  3726.702860
2           Married  Female      No    18.940352  ...  4193.190458
3           Widowed  Female      No    18.057507  ...  2434.234222
4           Married    Male      No    16.576858  ...  2127.830423
5           Widowed  Female      No    17.439069  ...  2113.073274

           Additional_charges  Timely admission Timely treatment  \
CaseOrder
1                17939.403420                 3                 3
2                17612.998120                 3                 4
3                17505.192460                 2                 4
4                12993.437350                 3                 5
5                 3716.525786                 2                 1

          Timely visits Reliability Options Hours of treatment  \
CaseOrder
1                     2           2       4                  3
2                     3           4       4                  4
3                     4           4       3                  4
4                     5           3       4                  5
5                     3           3       5                  3

          Courteous staff Evidence of active listening
CaseOrder
1                       3                           4
2                       3                           3
3                       3                           3
4                       5                           5
5                       4                           3

[5 rows x 39 columns]
           Children           Age         Income   VitD_levels    Doc_visits  \
count  10000.000000  10000.000000   10000.000000  10000.000000  10000.000000
mean       2.097200     53.511700   40490.495160     17.964262      5.012200
std        2.163659     20.638538   28521.153293      2.017231      1.045734
```

| | | | | | |
|---|---|---|---|---|---|
| min | 0.000000 | 18.000000 | 154.080000 | 9.806483 | 1.000000 |
| 25% | 0.000000 | 36.000000 | 19598.775000 | 16.626439 | 4.000000 |
| 50% | 1.000000 | 53.000000 | 33768.420000 | 17.951122 | 5.000000 |
| 75% | 3.000000 | 71.000000 | 54296.402500 | 19.347963 | 6.000000 |
| max | 10.000000 | 89.000000 | 207249.100000 | 26.394449 | 9.000000 |

| | Full_meals_eaten | vitD_supp | Initial_days | TotalCharge | \ |
|---|---|---|---|---|---|
| count | 10000.000000 | 10000.000000 | 10000.000000 | 10000.000000 | |
| mean | 1.001400 | 0.398900 | 34.455299 | 5312.172769 | |
| std | 1.008117 | 0.628505 | 26.309341 | 2180.393838 | |
| min | 0.000000 | 0.000000 | 1.001981 | 1938.312067 | |
| 25% | 0.000000 | 0.000000 | 7.896215 | 3179.374015 | |
| 50% | 1.000000 | 0.000000 | 35.836244 | 5213.952000 | |
| 75% | 2.000000 | 1.000000 | 61.161020 | 7459.699750 | |
| max | 7.000000 | 5.000000 | 71.981490 | 9180.728000 | |

| | Additional_charges | Timely admission | Timely treatment | Timely visits | \ |
|---|---|---|---|---|---|
| count | 10000.000000 | 10000.000000 | 10000.000000 | 10000.000000 | |
| mean | 12934.528587 | 3.518800 | 3.506700 | 3.511100 | |
| std | 6542.601544 | 1.031966 | 1.034825 | 1.032755 | |
| min | 3125.703000 | 1.000000 | 1.000000 | 1.000000 | |
| 25% | 7986.487755 | 3.000000 | 3.000000 | 3.000000 | |
| 50% | 11573.977735 | 4.000000 | 3.000000 | 4.000000 | |
| 75% | 15626.490000 | 4.000000 | 4.000000 | 4.000000 | |
| max | 30566.070000 | 8.000000 | 7.000000 | 8.000000 | |

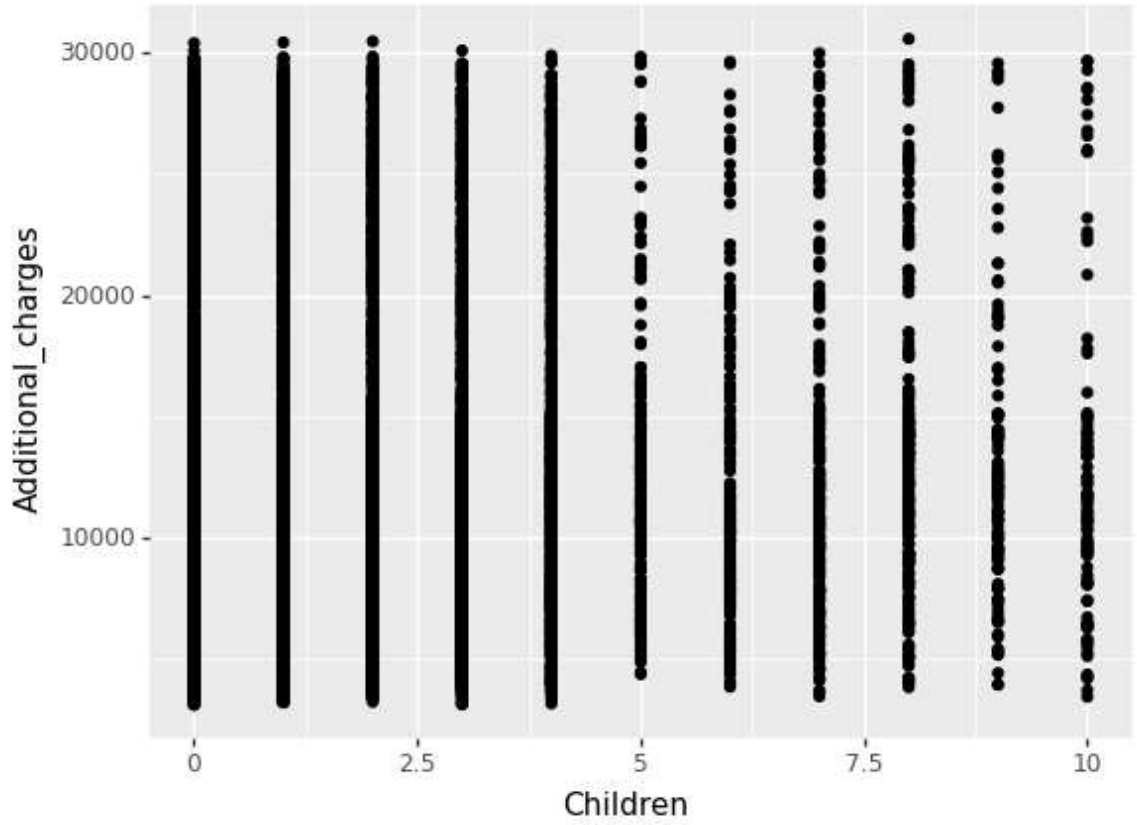
| | Reliability | Options | Hours of treatment | Courteous staff | \ |
|---|---|---|---|---|---|
| count | 10000.000000 | 10000.000000 | 10000.000000 | 10000.000000 | |
| mean | 3.515100 | 3.496900 | 3.522500 | 3.494000 | |
| std | 1.036282 | 1.030192 | 1.032376 | 1.021405 | |
| min | 1.000000 | 1.000000 | 1.000000 | 1.000000 | |
| 25% | 3.000000 | 3.000000 | 3.000000 | 3.000000 | |
| 50% | 4.000000 | 3.000000 | 4.000000 | 3.000000 | |
| 75% | 4.000000 | 4.000000 | 4.000000 | 4.000000 | |
| max | 7.000000 | 7.000000 | 7.000000 | 7.000000 | |

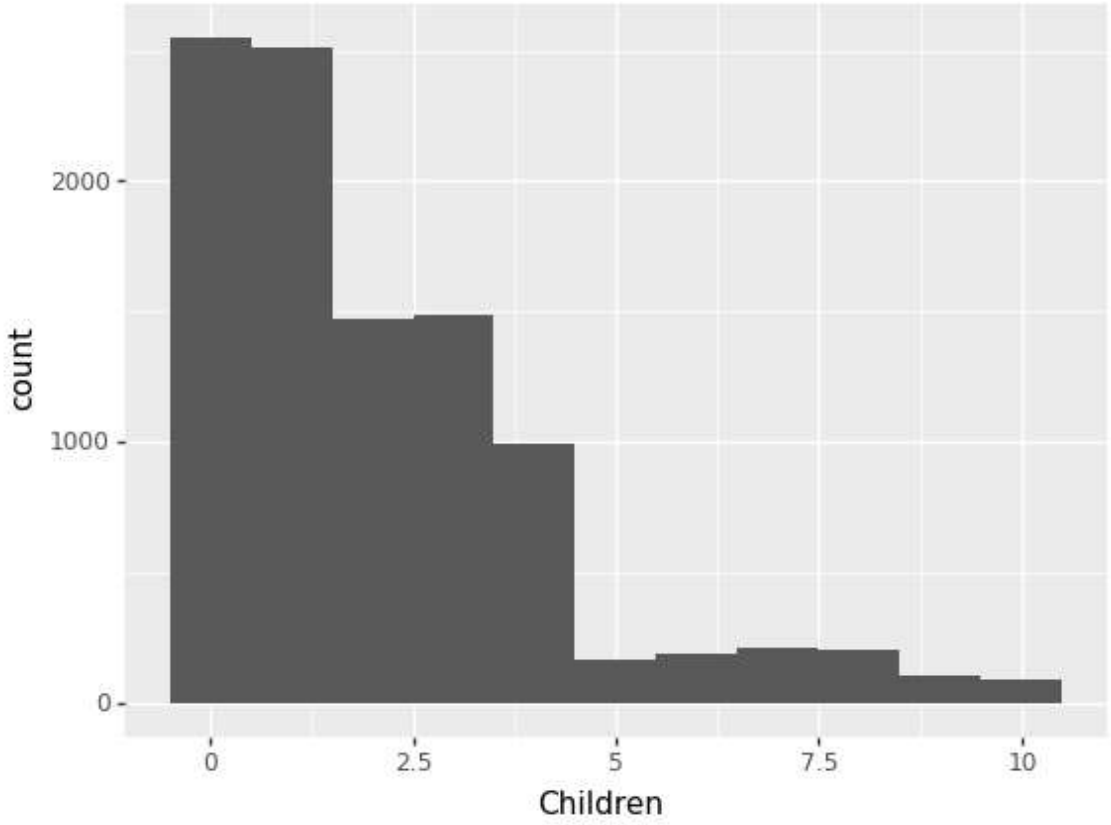
| | Evidence of active listening |
|---|---|
| count | 10000.000000 |
| mean | 3.509700 |
| std | 1.042312 |
| min | 1.000000 |
| 25% | 3.000000 |
| 50% | 3.000000 |
| 75% | 4.000000 |
| max | 7.000000 |

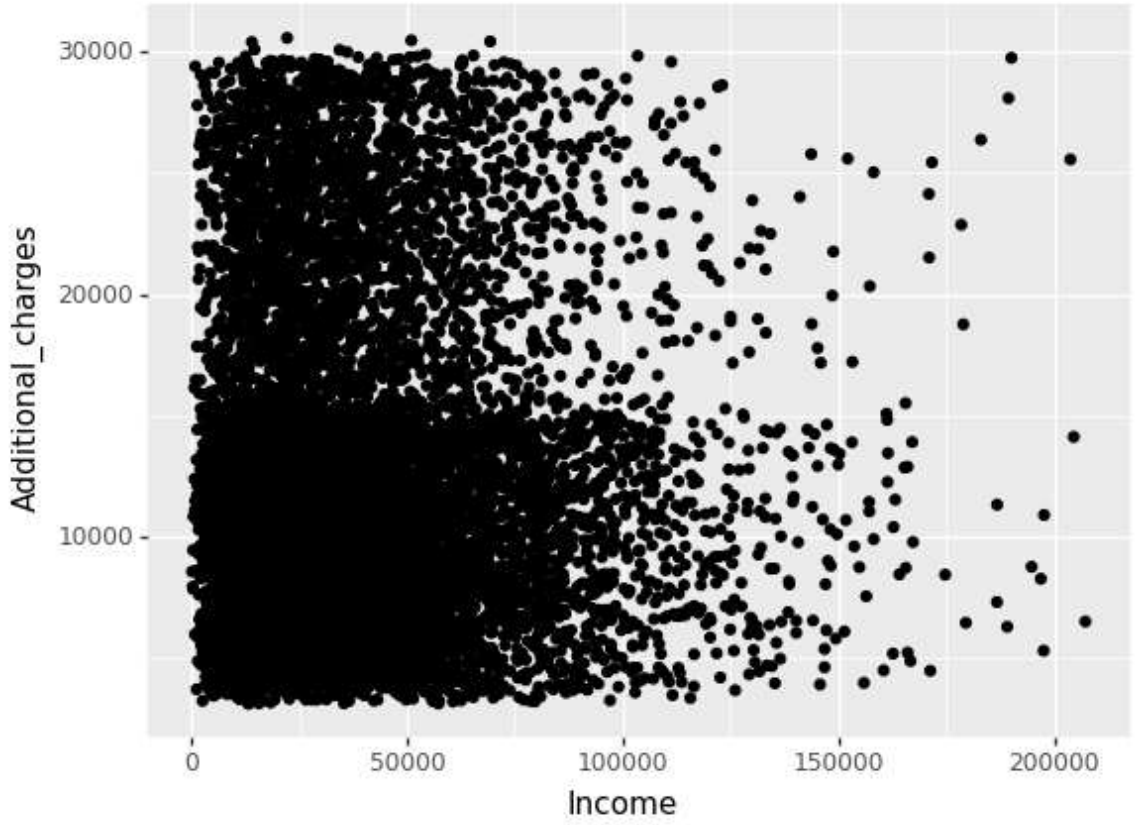
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 10000 entries, 1 to 10000
Data columns (total 39 columns):
 #   Column                        Non-Null Count  Dtype  
---  ------                        --------------  -----  
 0   State                         10000 non-null  object 
 1   Area                          10000 non-null  object 
 2   TimeZone                      10000 non-null  object 
 3   Children                      10000 non-null  int64  
 4   Age                           10000 non-null  int64  
 5   Income                        10000 non-null  float64
 6   Marital                       10000 non-null  object 
 7   Gender                        10000 non-null  object 
 8   ReAdmis                       10000 non-null  object 
 9   VitD_levels                   10000 non-null  float64
 10  Doc_visits                    10000 non-null  int64  
 11  Full_meals_eaten              10000 non-null  int64  
 12  vitD_supp                     10000 non-null  int64  
 13  Soft_drink                    10000 non-null  object 
 14  Initial_admin                 10000 non-null  object 
 15  HighBlood                     10000 non-null  object 
 16  Stroke                        10000 non-null  object 
 17  Complication_risk             10000 non-null  object 
 18  Overweight                    10000 non-null  object 
 19  Arthritis                     10000 non-null  object 
 20  Diabetes                      10000 non-null  object 
 21  Hyperlipidemia                10000 non-null  object 
 22  BackPain                      10000 non-null  object 
 23  Anxiety                       10000 non-null  object 
 24  Allergic_rhinitis             10000 non-null  object 
 25  Reflux_esophagitis            10000 non-null  object 
 26  Asthma                        10000 non-null  object 
 27  Services                      10000 non-null  object 
 28  Initial_days                  10000 non-null  float64
 29  TotalCharge                   10000 non-null  float64
 30  Additional_charges            10000 non-null  float64
 31  Timely admission              10000 non-null  int64  
 32  Timely treatment              10000 non-null  int64  
 33  Timely visits                 10000 non-null  int64  
 34  Reliability                   10000 non-null  int64  
 35  Options                       10000 non-null  int64  
 36  Hours of treatment            10000 non-null  int64  
 37  Courteous staff               10000 non-null  int64  
 38  Evidence of active listening  10000 non-null  int64  
dtypes: float64(5), int64(13), object(21)
```

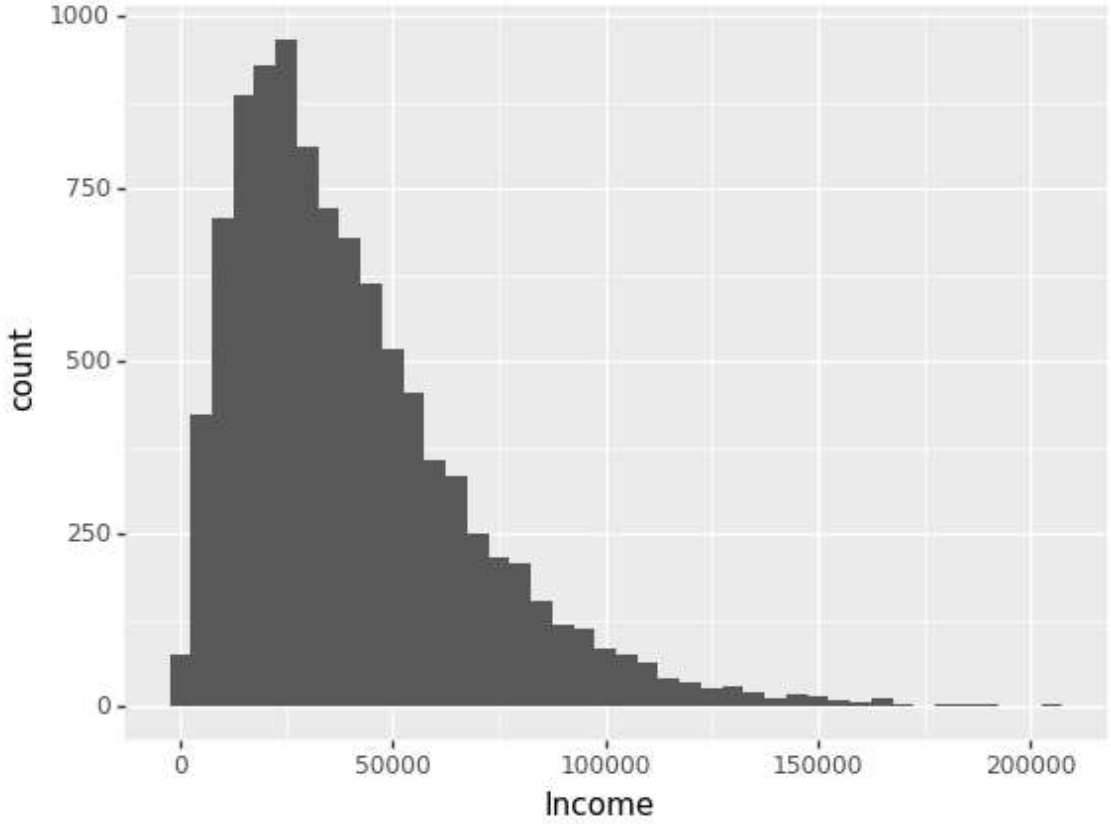
```
memory usage: 3.1+ MB
None
```

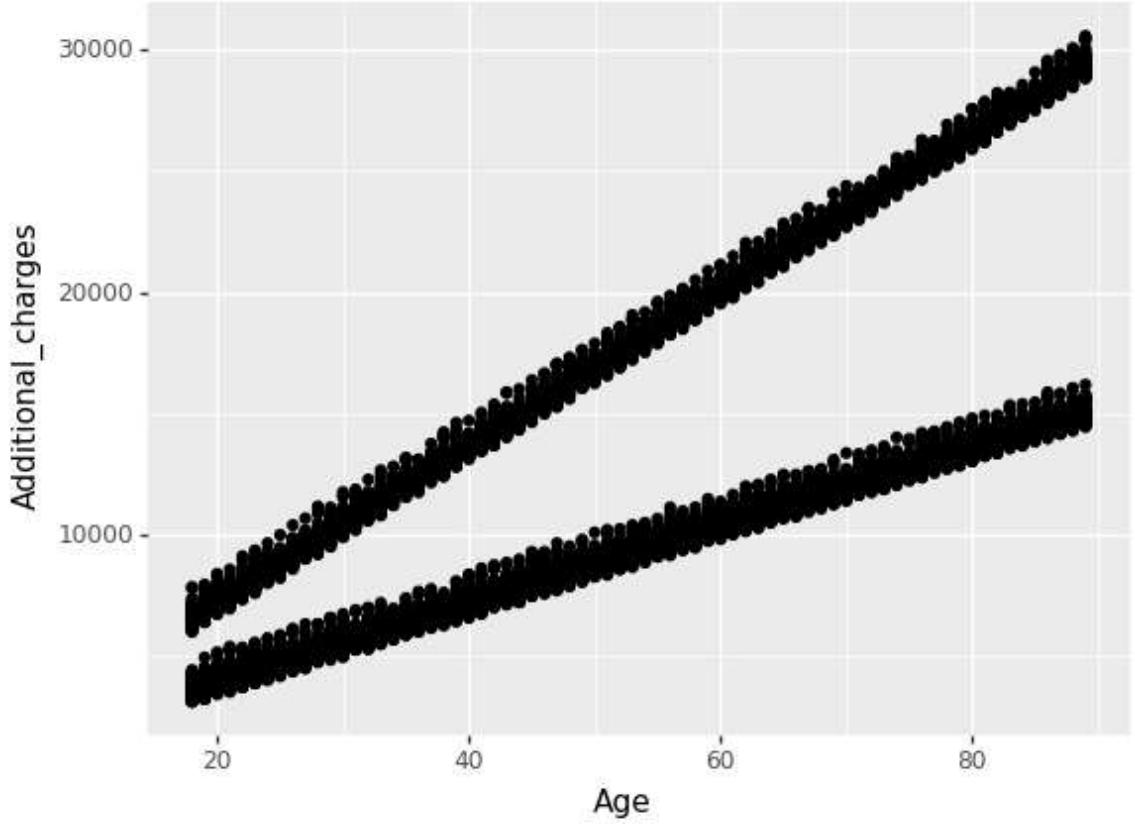
In [7]:
```
#Continuous Visualizations
#children
print(p9.ggplot(med_predict)+ p9.aes(x='Children',y='Additional_charges')+ p9.geom_point())
print(p9.ggplot(med_predict)+p9.aes(x='Children', fill='Children')+ p9.geom_histogram(binwidth=1))
#Income
print(p9.ggplot(med_predict)+ p9.aes(x='Income',y='Additional_charges')+ p9.geom_point())
print(p9.ggplot(med_predict)+p9.aes(x='Income', fill='Income')+ p9.geom_histogram(binwidth=5000))
#Age
print(p9.ggplot(med_predict)+ p9.aes(x='Age',y='Additional_charges')+ p9.geom_point())
print(p9.ggplot(med_predict)+p9.aes(x='Age', fill='Age')+ p9.geom_histogram(binwidth=5))
#Additional Charges
print(p9.ggplot(med_predict)+p9.aes(x='Additional_charges', fill='Additional_charges')+ p9.geom_histogram(binwidth=1000))
```

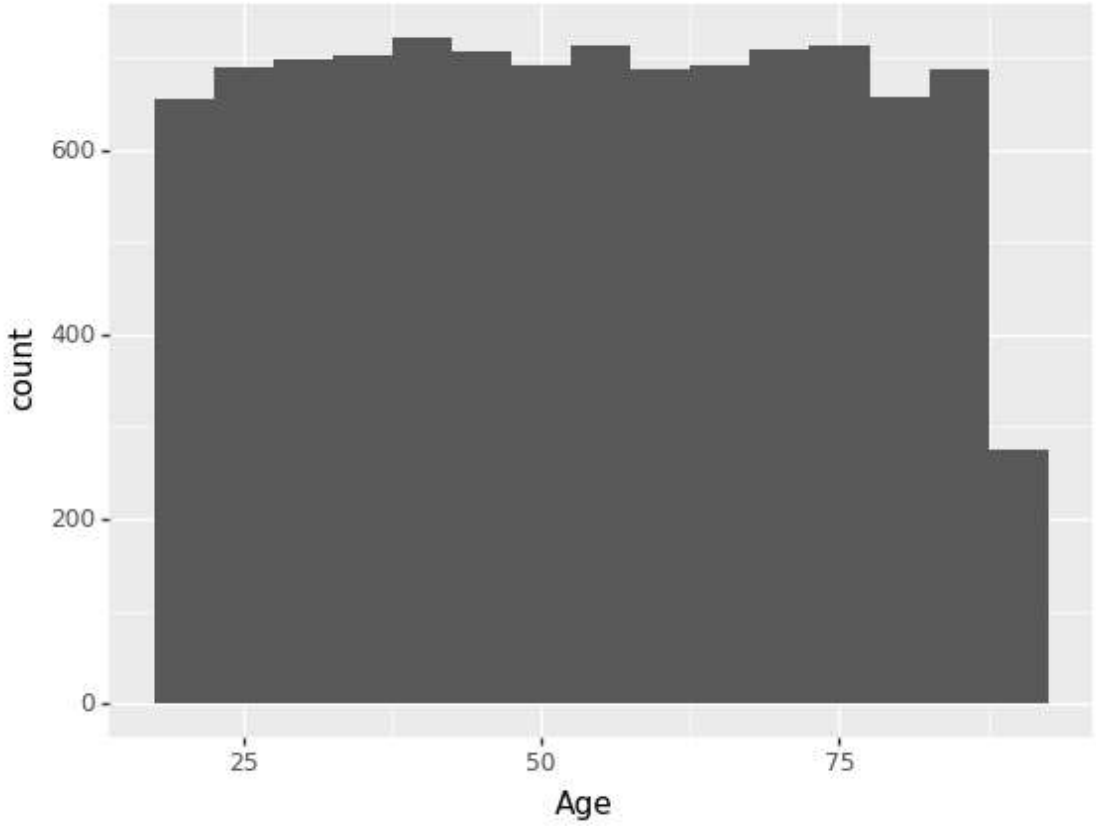


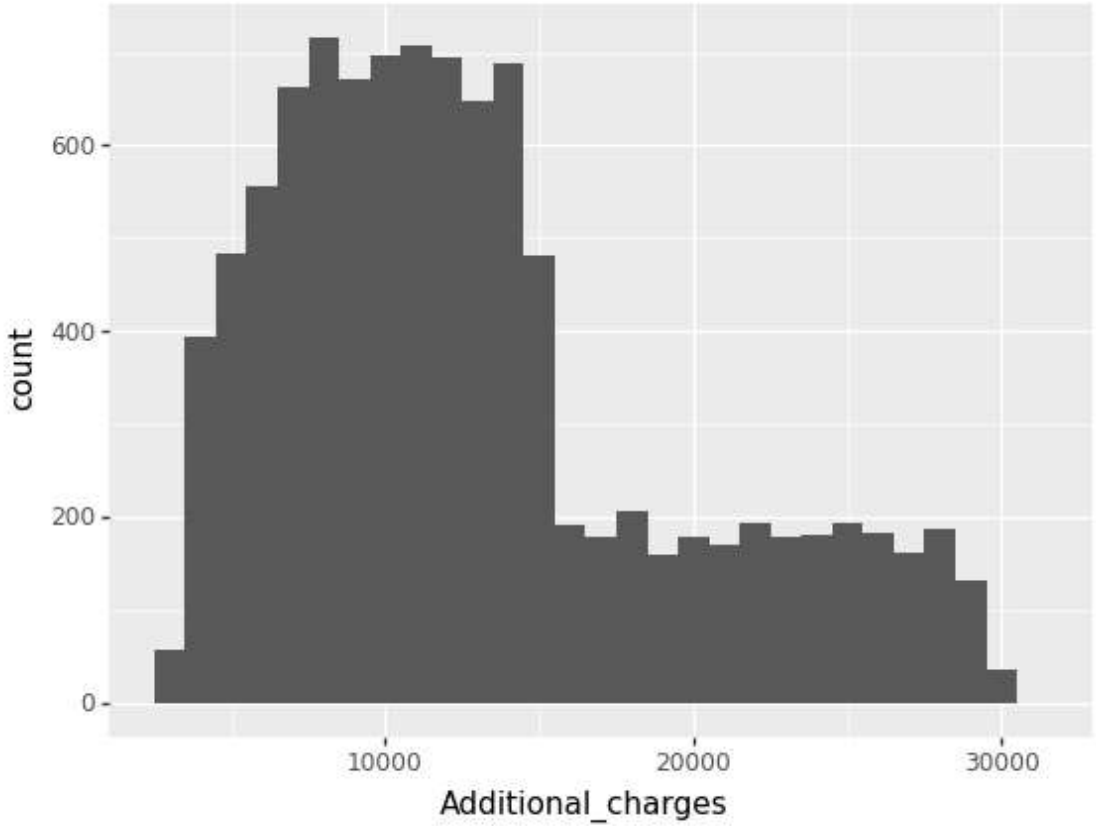

2000
1000
0
count
0
2.5
5
7.5
10
Children


30000
20000
10000
Additional_charges
0
50000
100000
150000
200000
Income


1000
750
500
250
0
count
0
50000
100000
150000
200000
Income


30000
20000
10000
Additional_charges
20
40
60
80
Age


600
400
200
0
count
25
50
75
Age




In [8]:
```
#Generate mean values by category to view relationship
print('Mean values of Additional Charges by category: ')
#complicationrisk
complicatedsummarystats=med_predict.groupby("Complication_risk")["Additional_charges"].mean()
print(complicatedsummarystats)
#Gender
gendersummarystats=med_predict.groupby("Gender")["Additional_charges"].mean()
print(gendersummarystats)
#High Blood Pressure
highbloodsummarystats=med_predict.groupby("HighBlood")["Additional_charges"].mean()
print(highbloodsummarystats)
#Services
servicessummarystats=med_predict.groupby("Services")["Additional_charges"].mean()
print(servicessummarystats)
```

```
Mean values of Additional Charges by category:
Complication_risk
High      13306.659302
Low       12490.777547
Medium    12866.641942
Name: Additional_charges, dtype: float64
Gender
Female       12896.066069
Male         12953.426237
Nonbinary    13415.374018
Name: Additional_charges, dtype: float64
HighBlood
No      9373.428702
Yes    18080.274388
Name: Additional_charges, dtype: float64
Services
Blood Work     12863.241599
CT Scan        13164.548730
Intravenous    12923.916677
MRI            13268.124891
Name: Additional_charges, dtype: float64
```

In [9]:
```
#Export Cleaned data to a CSV file
med_predict.to_csv('/Users/herlihpj/Desktop/Data Analytics/D208 - Predictive Modeling/Task 1/Medical/Prepared_data.csv')
```

# Data Analysis

In [10]:
```
#Function that builds multiple linear regression formula
def twoway_formula_builder(target_variable, predictor_variables):
    formula=target_variable+' ~ ('
    end=len(predictor_variables)-1
    #use enumerate to get first and last for string manipulation
    for count, value in enumerate(predictor_variables):
        if count==end:
            formula=formula+value
        else:
            formula=formula+value+' + '
    formula=formula+') ** 2 + 0'
    return formula
```

In [11]:
```
predictors=['Children', 'Complication_risk', 'Age', 'Income', 'Gender', 'Soft_drink', 'HighBlood', 'Services']
target='Additional_charges'
print(twoway_formula_builder(target,predictors))
```

```
Additional_charges ~ (Children + Complication_risk + Age + Income + Gender + Soft_drink + HighBlood + Services) ** 2 + 0
```

## Construct initial model using all potential predictors

In [12]:
```
#Initial Model
i_mdl_addcharge=ols(twoway_formula_builder(target,predictors),data=med_predict).fit()
print(i_mdl_addcharge.summary())
```

OLS Regression Results

| | | | |
|---|---|---|---|
| Dep. Variable: | Additional_charges | R-squared: | 0.998 |
| Model: | OLS | Adj. R-squared: | 0.998 |
| Method: | Least Squares | F-statistic: | 5.837e+04 |
| Date: | Thu, 23 Mar 2023 | Prob (F-statistic): | 0.00 |
| Time: | 21:05:37 | Log-Likelihood: | -71728. |
| No. Observations: | 10000 | AIC: | 1.436e+05 |
| Df Residuals: | 9926 | BIC: | 1.441e+05 |
| Df Model: | 73 | | |
| Covariance Type: | nonrobust | | |

| | coef | std err | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| Complication_risk[High] | 1010.4330 | 30.694 | 32.920 | 0.000 | 950.267 | 1070.599 |
| Complication_risk[Low] | 442.4044 | 33.700 | 13.128 | 0.000 | 376.347 | 508.462 |
| Complication_risk[Medium] | 553.4560 | 29.255 | 18.918 | 0.000 | 496.109 | 610.803 |
| Gender[T.Male] | 73.0416 | 24.483 | 2.983 | 0.003 | 25.049 | 121.034 |
| Gender[T.Nonbinary] | -35.9585 | 89.242 | -0.403 | 0.687 | -210.891 | 138.974 |
| Soft_drink[T.Yes] | 11.1488 | 28.486 | 0.391 | 0.696 | -44.690 | 66.987 |
| HighBlood[T.Yes] | 97.8884 | 25.019 | 3.913 | 0.000 | 48.847 | 146.930 |
| Services[T.CT Scan] | 32.2946 | 39.947 | 0.808 | 0.419 | -46.010 | 110.599 |
| Services[T.Intravenous] | -2.4392 | 27.455 | -0.089 | 0.929 | -56.257 | 51.379 |
| Services[T.MRI] | -21.1196 | 64.717 | -0.326 | 0.744 | -147.978 | 105.739 |
| Complication_risk[T.Low]:Gender[T.Male] | 2.7624 | 17.801 | 0.155 | 0.877 | -32.131 | 37.656 |
| Complication_risk[T.Medium]:Gender[T.Male] | 15.1693 | 14.635 | 1.037 | 0.300 | -13.517 | 43.856 |
| Complication_risk[T.Low]:Gender[T.Nonbinary] | -37.6269 | 65.263 | -0.577 | 0.564 | -165.556 | 90.302 |
| Complication_risk[T.Medium]:Gender[T.Nonbinary] | -0.5522 | 51.026 | -0.011 | 0.991 | -100.573 | 99.469 |
| Complication_risk[T.Low]:Soft_drink[T.Yes] | 6.6426 | 20.290 | 0.327 | 0.743 | -33.131 | 46.416 |
| Complication_risk[T.Medium]:Soft_drink[T.Yes] | -14.2232 | 16.556 | -0.859 | 0.390 | -46.676 | 18.229 |
| Complication_risk[T.Low]:HighBlood[T.Yes] | 14.1983 | 18.052 | 0.787 | 0.432 | -21.187 | 49.584 |
| Complication_risk[T.Medium]:HighBlood[T.Yes] | 8.2092 | 14.680 | 0.559 | 0.576 | -20.566 | 36.984 |
| Complication_risk[T.Low]:Services[T.CT Scan] | -7.2985 | 28.346 | -0.257 | 0.797 | -62.862 | 48.265 |
| Complication_risk[T.Medium]:Services[T.CT Scan] | -23.6214 | 22.715 | -1.040 | 0.298 | -68.147 | 20.904 |
| Complication_risk[T.Low]:Services[T.Intravenous] | 40.8016 | 19.791 | 2.062 | 0.039 | 2.006 | 79.597 |
| Complication_risk[T.Medium]:Services[T.Intravenous] | 24.7887 | 16.332 | 1.518 | 0.129 | -7.226 | 56.803 |
| Complication_risk[T.Low]:Services[T.MRI] | -29.2281 | 47.723 | -0.612 | 0.540 | -122.776 | 64.320 |
| Complication_risk[T.Medium]:Services[T.MRI] | -26.1502 | 39.626 | -0.660 | 0.509 | -103.825 | 51.525 |
| Gender[T.Male]:Soft_drink[T.Yes] | -22.9048 | 14.685 | -1.560 | 0.119 | -51.690 | 5.880 |
| Gender[T.Nonbinary]:Soft_drink[T.Yes] | -18.8881 | 50.199 | -0.376 | 0.707 | -117.288 | 79.512 |
| Gender[T.Male]:HighBlood[T.Yes] | -6.4442 | 13.061 | -0.493 | 0.622 | -32.046 | 19.158 |
| Gender[T.Nonbinary]:HighBlood[T.Yes] | 31.2871 | 45.373 | 0.690 | 0.490 | -57.653 | 120.228 |
| Gender[T.Male]:Services[T.CT Scan] | 32.2366 | 20.379 | 1.582 | 0.114 | -7.711 | 72.184 |
| Gender[T.Nonbinary]:Services[T.CT Scan] | 39.5169 | 71.066 | 0.556 | 0.578 | -99.787 | 178.821 |
| Gender[T.Male]:Services[T.Intravenous] | -12.4172 | 14.473 | -0.858 | 0.391 | -40.787 | 15.953 |

| | | | | | | |
|---|---|---|---|---|---|---|
| Gender[T.Nonbinary]:Services[T.Intravenous] | -17.5530 | 50.938 | -0.345 | 0.730 | -117.403 | 82.297 |
| Gender[T.Male]:Services[T.MRI] | 10.3649 | 34.100 | 0.304 | 0.761 | -56.478 | 77.208 |
| Gender[T.Nonbinary]:Services[T.MRI] | -44.1599 | 230.252 | -0.192 | 0.848 | -495.500 | 407.180 |
| Soft_drink[T.Yes]:HighBlood[T.Yes] | 2.9741 | 14.823 | 0.201 | 0.841 | -26.082 | 32.030 |
| Soft_drink[T.Yes]:Services[T.CT Scan] | 8.3300 | 23.232 | 0.359 | 0.720 | -37.210 | 53.870 |
| Soft_drink[T.Yes]:Services[T.Intravenous] | 28.9975 | 16.360 | 1.772 | 0.076 | -3.071 | 61.066 |
| Soft_drink[T.Yes]:Services[T.MRI] | 63.9421 | 37.560 | 1.702 | 0.089 | -9.684 | 137.568 |
| HighBlood[T.Yes]:Services[T.CT Scan] | 8.8175 | 20.424 | 0.432 | 0.666 | -31.217 | 48.852 |
| HighBlood[T.Yes]:Services[T.Intravenous] | -12.3125 | 14.596 | -0.844 | 0.399 | -40.923 | 16.299 |
| HighBlood[T.Yes]:Services[T.MRI] | -5.5153 | 34.535 | -0.160 | 0.873 | -73.211 | 62.180 |
| Children | 40.0355 | 5.657 | 7.077 | 0.000 | 28.946 | 51.125 |
| Children:Complication_risk[T.Low] | 3.2962 | 4.071 | 0.810 | 0.418 | -4.684 | 11.277 |
| Children:Complication_risk[T.Medium] | 3.1451 | 3.337 | 0.943 | 0.346 | -3.395 | 9.686 |
| Children:Gender[T.Male] | -3.1327 | 2.971 | -1.054 | 0.292 | -8.957 | 2.691 |
| Children:Gender[T.Nonbinary] | -14.3002 | 10.375 | -1.378 | 0.168 | -34.638 | 6.037 |
| Children:Soft_drink[T.Yes] | -2.4131 | 3.352 | -0.720 | 0.472 | -8.983 | 4.157 |
| Children:HighBlood[T.Yes] | -0.3648 | 2.991 | -0.122 | 0.903 | -6.228 | 5.498 |
| Children:Services[T.CT Scan] | 3.0266 | 4.705 | 0.643 | 0.520 | -6.197 | 12.250 |
| Children:Services[T.Intravenous] | -2.4809 | 3.318 | -0.748 | 0.455 | -8.984 | 4.022 |
| Children:Services[T.MRI] | 11.9881 | 7.667 | 1.563 | 0.118 | -3.042 | 27.018 |
| Age | 160.3041 | 0.458 | 350.390 | 0.000 | 159.407 | 161.201 |
| Complication_risk[T.Low]:Age | 0.1071 | 0.424 | 0.253 | 0.801 | -0.724 | 0.938 |
| Complication_risk[T.Medium]:Age | 0.4805 | 0.351 | 1.368 | 0.171 | -0.208 | 1.169 |
| Age:Gender[T.Male] | 0.2058 | 0.311 | 0.661 | 0.508 | -0.404 | 0.816 |
| Age:Gender[T.Nonbinary] | -0.1289 | 1.088 | -0.118 | 0.906 | -2.262 | 2.004 |
| Age:Soft_drink[T.Yes] | 0.0164 | 0.352 | 0.047 | 0.963 | -0.673 | 0.706 |
| Age:HighBlood[T.Yes] | 159.5687 | 0.313 | 509.510 | 0.000 | 158.955 | 160.183 |
| Age:Services[T.CT Scan] | -0.4283 | 0.501 | -0.856 | 0.392 | -1.409 | 0.553 |
| Age:Services[T.Intravenous] | -0.1286 | 0.346 | -0.372 | 0.710 | -0.807 | 0.550 |
| Age:Services[T.MRI] | 0.0428 | 0.844 | 0.051 | 0.960 | -1.612 | 1.698 |
| Income | -0.0001 | 0.000 | -0.289 | 0.772 | -0.001 | 0.001 |
| Complication_risk[T.Low]:Income | 3.44e-05 | 0.000 | 0.110 | 0.913 | -0.001 | 0.001 |
| Complication_risk[T.Medium]:Income | 1.557e-05 | 0.000 | 0.061 | 0.951 | -0.000 | 0.001 |
| Income:Gender[T.Male] | 2.701e-05 | 0.000 | 0.120 | 0.905 | -0.000 | 0.000 |
| Income:Gender[T.Nonbinary] | 0.0015 | 0.001 | 2.074 | 0.038 | 8.26e-05 | 0.003 |
| Income:Soft_drink[T.Yes] | 0.0003 | 0.000 | 1.244 | 0.214 | -0.000 | 0.001 |
| Income:HighBlood[T.Yes] | -0.0004 | 0.000 | -1.628 | 0.104 | -0.001 | 7.58e-05 |
| Income:Services[T.CT Scan] | -0.0003 | 0.000 | -0.853 | 0.394 | -0.001 | 0.000 |
| Income:Services[T.Intravenous] | -0.0001 | 0.000 | -0.495 | 0.621 | -0.001 | 0.000 |
| Income:Services[T.MRI] | -0.0001 | 0.001 | -0.194 | 0.846 | -0.001 | 0.001 |
| Children:Age | -0.0581 | 0.071 | -0.821 | 0.411 | -0.197 | 0.081 |
| Children:Income | -8.714e-05 | 4.99e-05 | -1.748 | 0.081 | -0.000 | 1.06e-05 |
| Age:Income | 3.941e-06 | 5.39e-06 | 0.731 | 0.465 | -6.63e-06 | 1.45e-05 |

| | | | |
|---|---|---|---|
| Omnibus: | 1490.623 | Durbin-Watson: | 2.034 |

```
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2239.875
Skew:                           1.111   Prob(JB):                         0.00
Kurtosis:                       3.662   Cond. No.                     2.06e+08
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.06e+08. This might indicate that there are
strong multicollinearity or other numerical problems.
```

After reviewing the initial model summary of statistics, the P-values for HighBlood, Age, and the interaction between Highblood and Age have a statistical significance (0.000) on the variance in the Additional_charges variable. The remaining variables did not seem to have a strong correlation with the Additional_charges and therefore could be removed in the construction of the final model.

## Construct final model

In [13]:
```
#Reduced model:
mdl_addcharge=ols("Additional_charges ~ Age * HighBlood + 0",data=med_predict).fit()
#returns intercept and slope?
print('Parameters: ')
print(mdl_addcharge.params)
print('Summary: ')
print(mdl_addcharge.summary())
```

```
Parameters:
HighBlood[No]           792.107136
HighBlood[Yes]          894.875100
Age                     160.732010
Age:HighBlood[T.Yes]    159.359630
dtype: float64
Summary:
                            OLS Regression Results
==============================================================================
Dep. Variable:     Additional_charges   R-squared:                       0.996
Model:                            OLS   Adj. R-squared:                  0.996
Method:                 Least Squares   F-statistic:                 9.207e+05
Date:                Thu, 23 Mar 2023   Prob (F-statistic):               0.00
Time:                        21:06:02   Log-Likelihood:                -73924.
No. Observations:               10000   AIC:                         1.479e+05
Df Residuals:                    9996   BIC:                         1.479e+05
Df Model:                           3
Covariance Type:            nonrobust
===========================================================================================
                              coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------------
HighBlood[No]          792.1071     14.168     55.907      0.000     764.335     819.880
HighBlood[Yes]         894.8751     17.140     52.209      0.000     861.277     928.473
Age                    160.7320      0.248    649.405      0.000     160.247     161.217
Age:HighBlood[T.Yes]   159.3596      0.387    411.351      0.000     158.600     160.119
==============================================================================
Omnibus:                      632.697   Durbin-Watson:                   2.034
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              759.054
Skew:                           0.673   Prob(JB):                     1.49e-165
Kurtosis:                       3.107   Cond. No.                         275.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

In [14]:
```
all_resid=mdl_addcharge.resid
print(all_resid.sum()) #Sum of residuals should be ~0
```

```
-5.568908818531781e-07
```

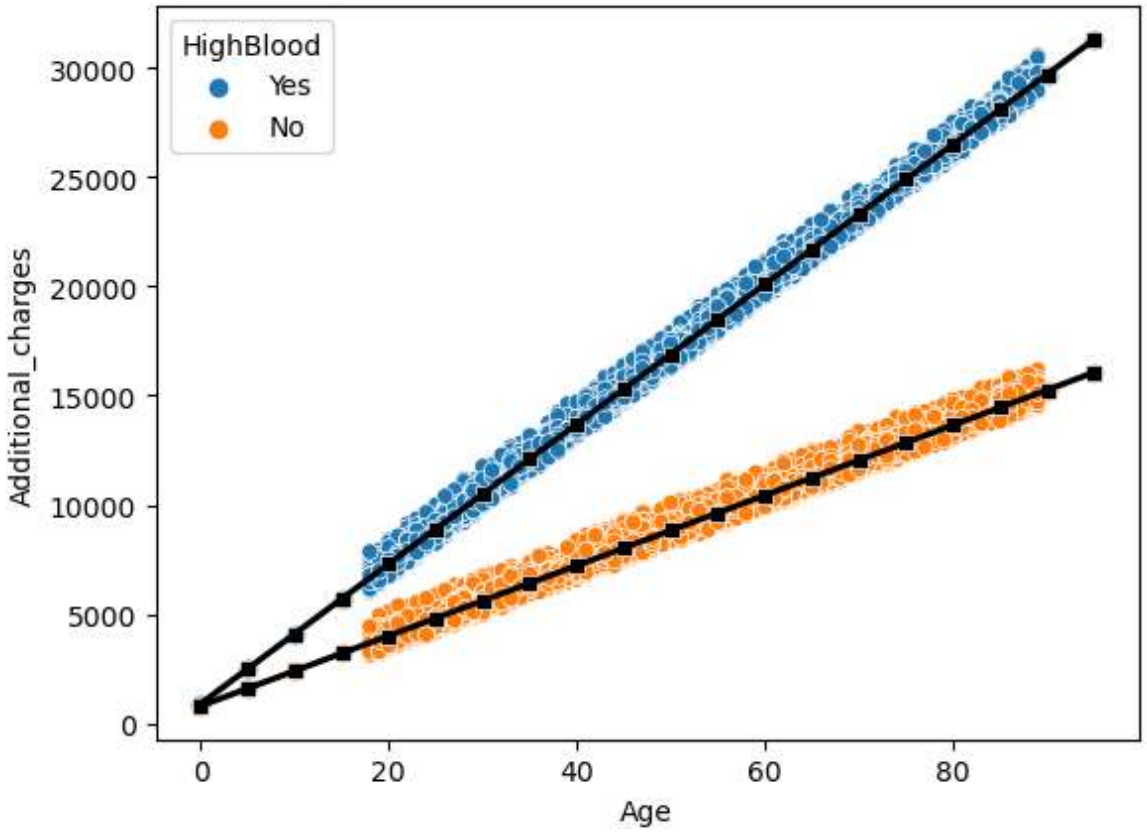
In [15]:
```
#Prediction Values
#Create exploratory data for the model
print('Prediction Values: ')
AGES = np.arange(0, 100, 5)
H_Blood=med_predict['HighBlood'].unique()
#creates an array of all possible 'Age' and 'HighBlood' outcomes for Age in 5 year increments
```

```
p=product(AGES,H_Blood)
#Create a dataframe with all the exploratory values
expl_data = pd.DataFrame(p,columns=["Age","HighBlood"])
#Adds the predicted values to the new data frame
pred_data = expl_data.assign(Additional_charges = mdl_addcharge.predict(expl_data))
print(pred_data)
```

```
Prediction Values:
    Age HighBlood  Additional_charges
0     0       Yes          894.875100
1     0        No          792.107136
2     5       Yes         2495.333301
3     5        No         1595.767187
4    10       Yes         4095.791502
5    10        No         2399.427238
6    15       Yes         5696.249703
7    15        No         3203.087289
8    20       Yes         7296.707904
9    20        No         4006.747340
10   25       Yes         8897.166105
11   25        No         4810.407391
12   30       Yes        10497.624306
13   30        No         5614.067443
14   35       Yes        12098.082507
15   35        No         6417.727494
16   40       Yes        13698.540708
17   40        No         7221.387545
18   45       Yes        15298.998909
19   45        No         8025.047596
20   50       Yes        16899.457110
21   50        No         8828.707647
22   55       Yes        18499.915311
23   55        No         9632.367698
24   60       Yes        20100.373512
25   60        No        10436.027749
26   65       Yes        21700.831713
27   65        No        11239.687801
28   70       Yes        23301.289913
29   70        No        12043.347852
30   75       Yes        24901.748114
31   75        No        12847.007903
32   80       Yes        26502.206315
33   80        No        13650.667954
34   85       Yes        28102.664516
35   85        No        14454.328005
36   90       Yes        29703.122717
37   90        No        15257.988056
38   95       Yes        31303.580918
39   95        No        16061.648107
```

In [16]:
```
#Visualize how the predicted values fit the data (Waskom)
sns.scatterplot(x="Age",y="Additional_charges",data=med_predict,hue='HighBlood')
```

```
sns.regplot(x="Age", y="Additional_charges", data=pred_data[pred_data.HighBlood=='Yes'], ci=None, line_kws={"color": "bla
sns.regplot(x="Age", y="Additional_charges", data=pred_data[pred_data.HighBlood=='No'], ci=None, line_kws={"color": "blac
sns.scatterplot(x="Age",y="Additional_charges",data=pred_data,color='black',legend=False,marker="s")
plt.show()
```



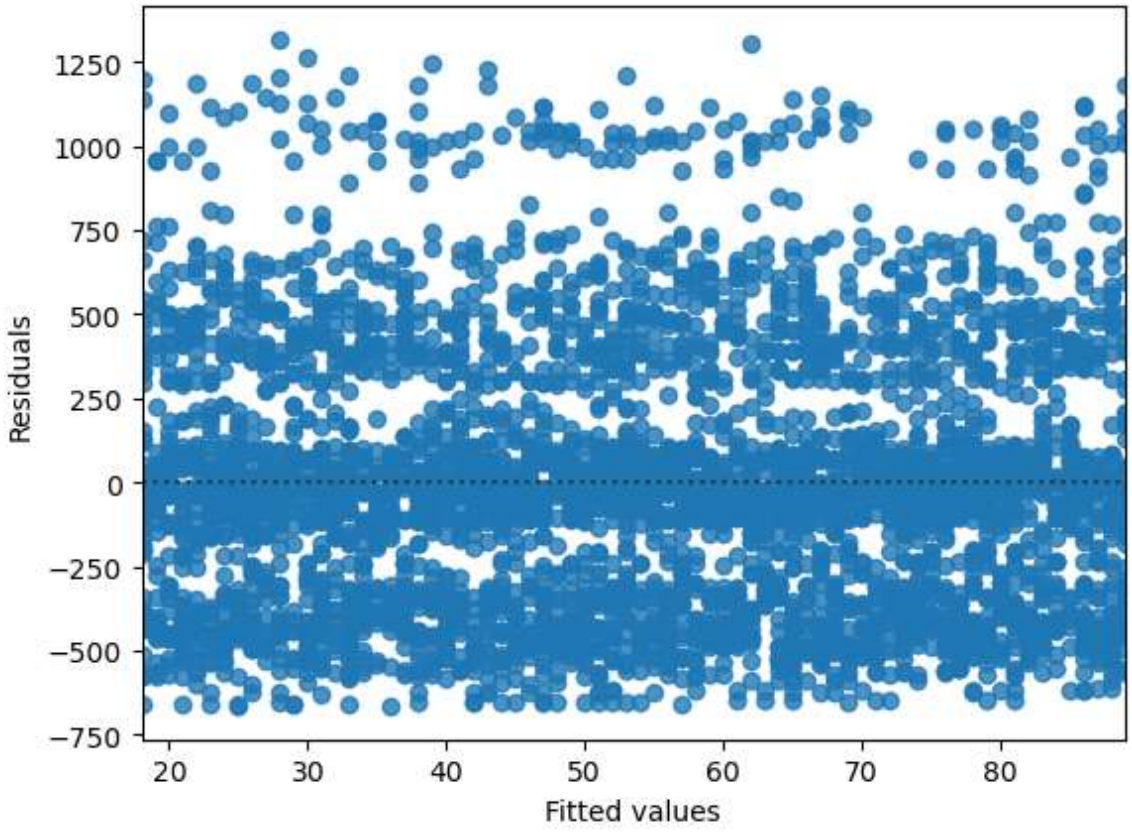

# Model Evaluation

In [17]:
```
#Check the model
residuals=mdl_addcharge.resid
R_squared=residuals**2
#calc RSE
mse=mdl_addcharge.mse_resid
rse=np.sqrt(mse)
print('RSE: ', rse)
```

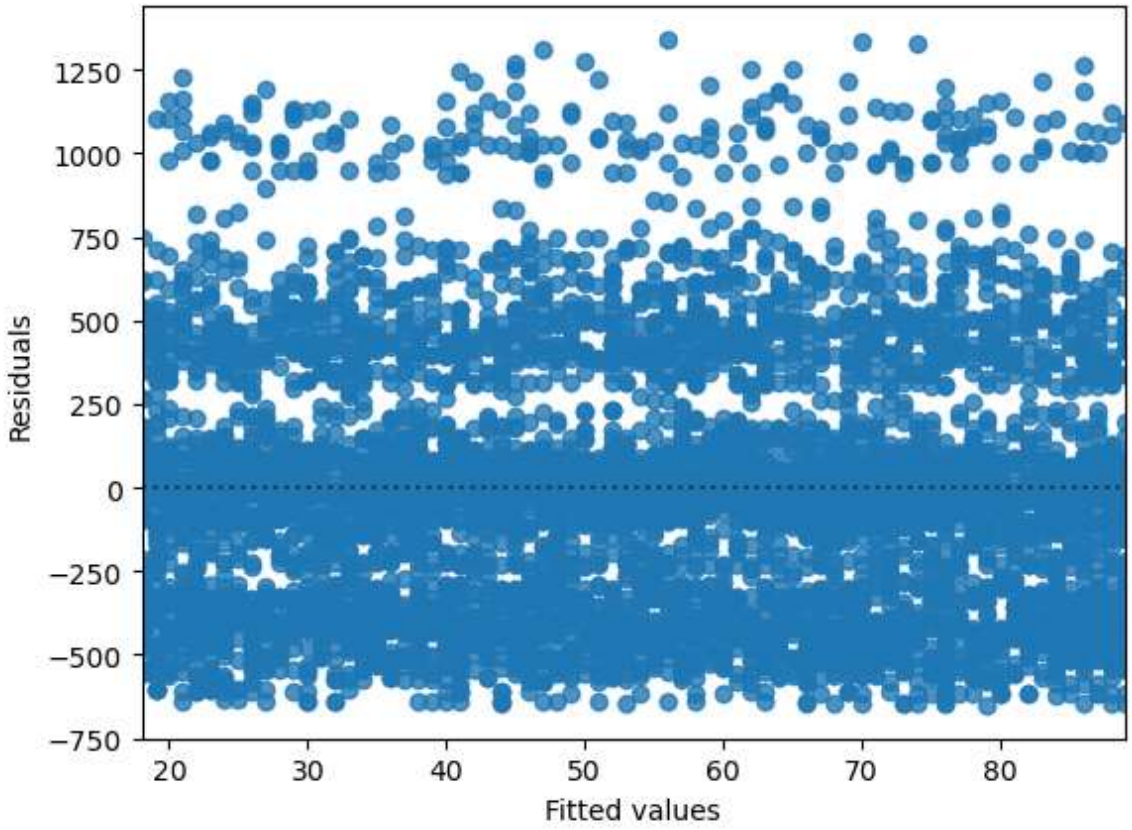
```
RSE:  392.94403461162545
```

In [18]:
```
#Residual Plots
print('HighBlood=Yes Residual Plot')
sns.residplot(x="Age", y="Additional_charges", data=med_predict[med_predict.HighBlood=='Yes'], lowess=True)
plt.xlabel("Fitted values")
plt.ylabel("Residuals")
plt.show()
print('HighBlood=No Residual Plot')
sns.residplot(x="Age", y="Additional_charges", data=med_predict[med_predict.HighBlood=='No'], lowess=True)
plt.xlabel("Fitted values")
plt.ylabel("Residuals")
plt.show()
```

```
HighBlood=Yes Residual Plot
```

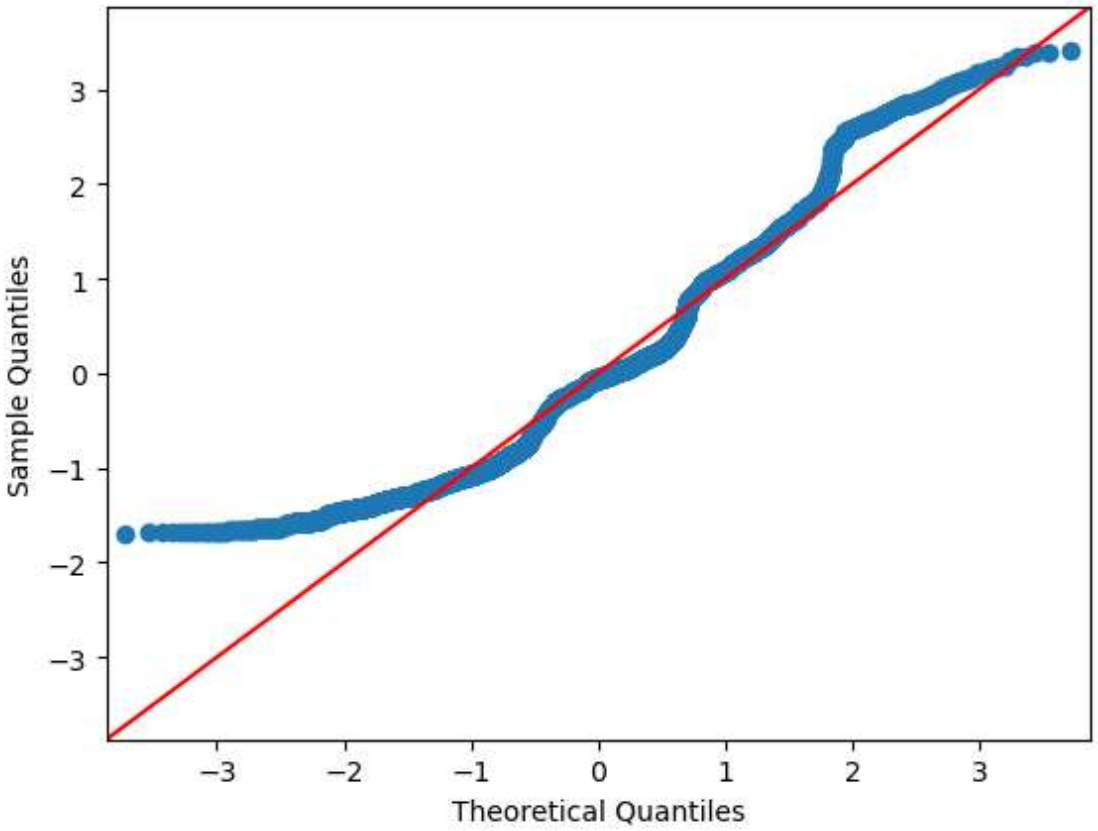



```
HighBlood=No Residual Plot
```




In [19]:
```
#Q-Q plot
qqplot(data=mdl_addcharge.resid, fit=True, line="45")
```

Out[19]:




In the initial and the final model the P-values for Age, HighBlood, and the interaction between Age and HighBlood were all 0.000 clearly indicating they have a major impact on the Additional_charge. When reducing the the initial model from 8 variables to 2 the R-Squared value only dropped from .998 to .996 showing that not much variablility was lost when dropping 6 variables.

Based on the results the hospital can now better predict additional costs for patients based on the patients age and whether or not they have high blood pressure.