Risk Factors and Insurance

November 15, 2020

0.1 Title: Why is Health Insurance taking money out of my Wallet?

0.2 Background and Field Research

According to the centers for Medicare and Medicaid Services: - Private health insurance spending grew 5.8 percent to 1,243 billion dollars in 2018. - Prescription drug spending increased 2.5 percent to 335.0 billion dollars in 2018, faster than the 1.4 growth in 2017 - Future predictions are projecting National health spending is projected to grow at an average annual rate of 5.4 percent for 2019-28 and to reach 6.2 trillion by 2028.

0.3 Buisness Objective

Using K-means clustering can a machine learning model accurately reveal the most contributing factor to the costs of health insurance?

0.4 Data Dictionary:

Column Name	Description
Age	Number of times pregnant
Gender	Male or Female
BMI	Body Mass Index in Kg
Number of Children	Total number of children
Charges	Amount of Insurance in dollars

1 Import Libraries

[18]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from matplotlib import style
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_samples, silhouette_score
from sklearn.preprocessing import StandardScaler
from matplotlib.ticker import MaxNLocator
from statsmodels.formula.api import ols
```

2 Load Data

[62]:
```
raw_data = pd.read_csv('insurance.csv')
```

3 Exploratory Data Analysis (EDA)

[71]:
```
raw_data.shape
```

[71]:
```
(1338, 7)
```

3.1 Summary Statistics

We see outliers in Charges but none in other classes

[22]:
```
raw_data.describe()
```

[22]:

	age	bmi	children	charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.663397	1.094918	13270.422265
std	14.049960	6.098187	1.205493	12110.011237
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.296250	0.000000	4740.287150
50%	39.000000	30.400000	1.000000	9382.033000
75%	51.000000	34.693750	2.000000	16639.912515
max	64.000000	53.130000	5.000000	63770.428010

3.2 Check for Null Values

[23]:
```
raw_data.isnull().sum()
```

[23]:
```
age         0
sex         0
bmi         0
children    0
smoker      0
region      0
charges     0
dtype: int64
```

3.3 Drop Columns

We are only interested in continous variables so reduce chance of error or noise by removing other columns

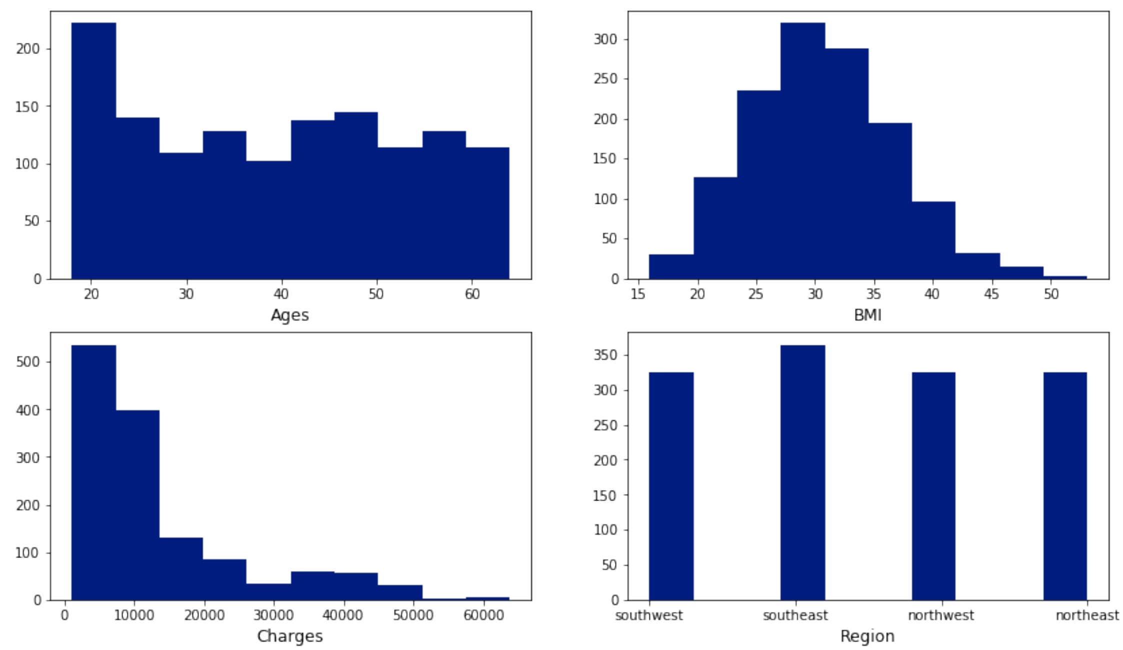
[21]:
```
raw_data_c=raw_data.drop(["sex", "smoker", "region"], axis=1).copy() #only␣
↪continuous variable dataset will be used for plots
```

3.4 Analyze Histogram

Another way to visualize whether there is any skewness in the data. As you can see it is skewed right for charges

[25]:
```
plt.figure(figsize=(14,8))
style.use("seaborn-dark-palette")
plt.subplot(2,2,1)
plt.hist(raw_data["age"])
plt.xlabel("Ages", fontsize=12)
plt.subplot(2,2,2)
plt.hist(raw_data["bmi"])
plt.xlabel("BMI", fontsize=12)
plt.subplot(2,2,3)
plt.hist(raw_data["charges"])
plt.xlabel("Charges", fontsize=12)
plt.subplot(2,2,4)
plt.hist(raw_data["region"])
plt.xlabel("Region", fontsize=12)
;
```

[25]: ''

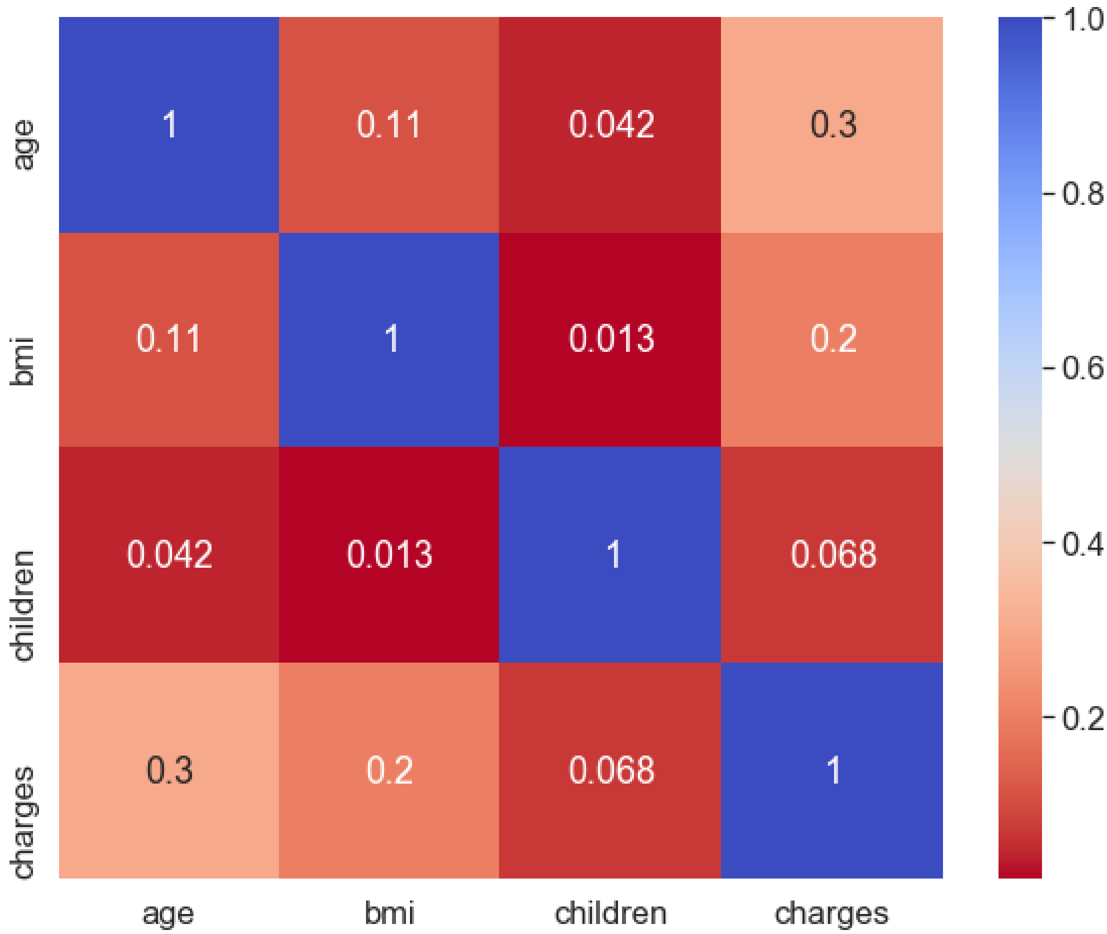


3.5 Correlation

From correlation plot we can see age and charges have very slight positive correlation with charges which we will try to prove in due course.

[27]:
```
plt.figure(figsize=(10,8))
corar=np.array(corr_mat.values)
sns.set(font_scale=1.5)
sns.heatmap(corr_mat, annot=corar,cmap="coolwarm_r")
```

[27]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fccbfbab850>
```



3.6 Feature Engineering

Convert the age into bins/groups of categorical variables like Child, Young Adult, Adult and Old to analyse relation with medical expenses "charges" - Less than 30 is young Adult - Between 30 and 59 is Adult - Over 59 is Old Adult

[28]:
```
raw_data.age.describe()
```

[28]:
```
count    1338.000000
mean       39.207025
std        14.049960
```

```
min          18.000000
25%          27.000000
50%          39.000000
75%          51.000000
max          64.000000
Name: age, dtype: float64
```

[29]:
```
raw_data.loc[(raw_data.age>17) & (raw_data.age<=30), "age_cat"]="Young Adult"
raw_data.loc[(raw_data.age>30) & (raw_data.age<=59), "age_cat"]="Adult"
raw_data.loc[(raw_data.age>59), "age_cat"]="Old"
raw_data
```

[29]:
```
      age     sex     bmi  children smoker     region      charges  \
0      19  female  27.900         0    yes  southwest  16884.92400
1      18    male  33.770         1     no  southeast   1725.55230
2      28    male  33.000         3     no  southeast   4449.46200
3      33    male  22.705         0     no  northwest  21984.47061
4      32    male  28.880         0     no  northwest   3866.85520
…     …      …       …        …     …         …          …
1333   50    male  30.970         3     no  northwest  10600.54830
1334   18  female  31.920         0     no  northeast   2205.98080
1335   18  female  36.850         0     no  southeast   1629.83350
1336   21  female  25.800         0     no  southwest   2007.94500
1337   61  female  29.070         0    yes  northwest  29141.36030

          age_cat
0     Young Adult
1     Young Adult
2     Young Adult
3           Adult
4           Adult
…             …
1333        Adult
1334  Young Adult
1335  Young Adult
1336  Young Adult
1337          Old

[1338 rows x 8 columns]
```

3.7 Count and Visualize unique values

- Lets see how many young adults, adults and old adults there are based on our feature engineering. The most is Adult with 780.
- The cirlce plot below provides as quantitative amount as a percentage. Adults are 33.2% of entire dataset

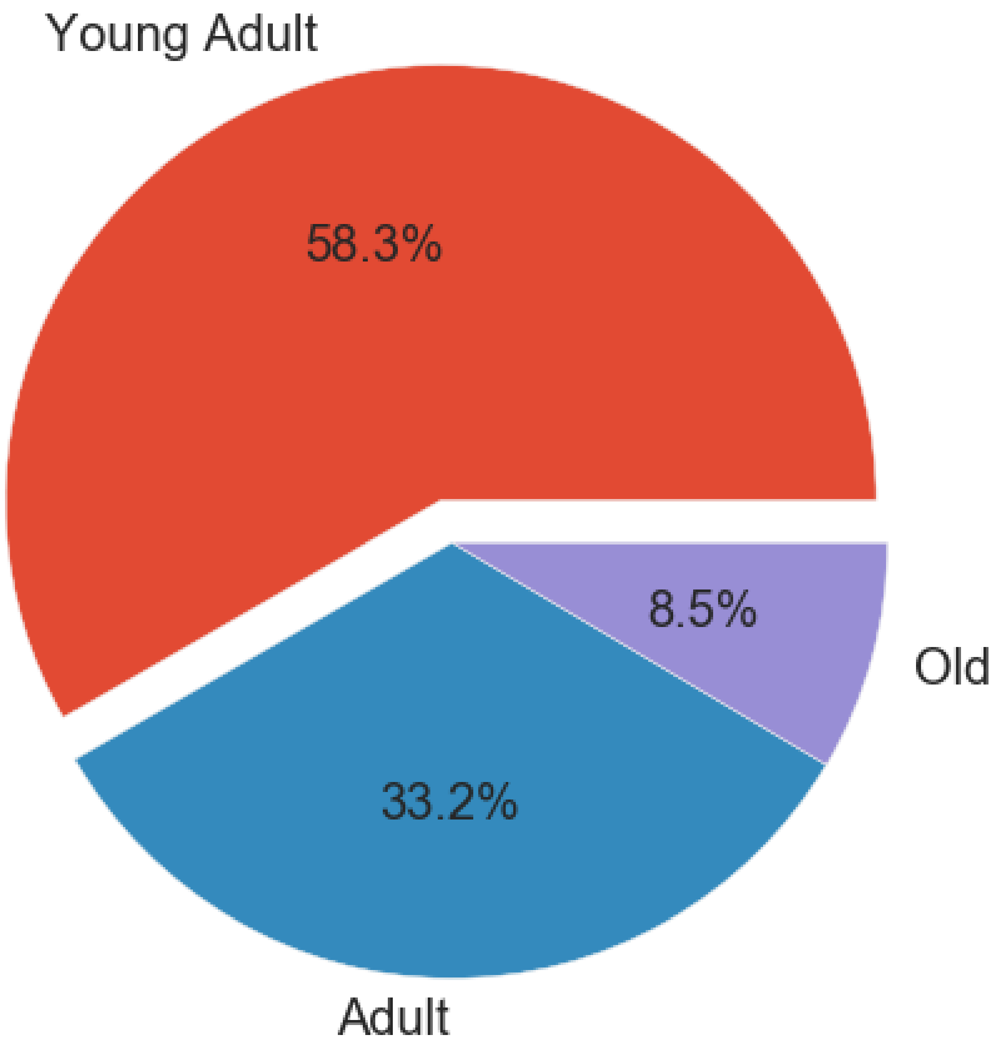
[30]:

```
labels=raw_data.age_cat.unique().tolist()
count=raw_data.age_cat.value_counts()
print(count)
count=count.values
style.use("ggplot")
plt.figure(figsize=(8,8))
explode=(0.1,0,0)
plt.pie(count, labels=labels,explode=explode, autopct="%1.1f%%",␣
↪textprops={'fontsize': 20})
```

```
Adult           780
Young Adult     444
Old             114
Name: age_cat, dtype: int64
```

[30]:

```
([<matplotlib.patches.Wedge at 0x7fccc0154150>,
  <matplotlib.patches.Wedge at 0x7fccc0154890>,
  <matplotlib.patches.Wedge at 0x7fccc015c090>],
 [Text(-0.30922189662362254, 1.159474802938162, 'Young Adult'),
  Text(-0.007748139924787676, -1.0999727116286595, 'Adult'),
  Text(1.0608289775377782, -0.29093277645557974, 'Old')],
 [Text(-0.18037943969711315, 0.6763603017139278, '58.3%'),
  Text(-0.004226258140793277, -0.5999851154338142, '33.2%'),
  Text(0.5786339877478789, -0.1586906053394071, '8.5%')])
```



3.8 Group by Adult catagory and Charge

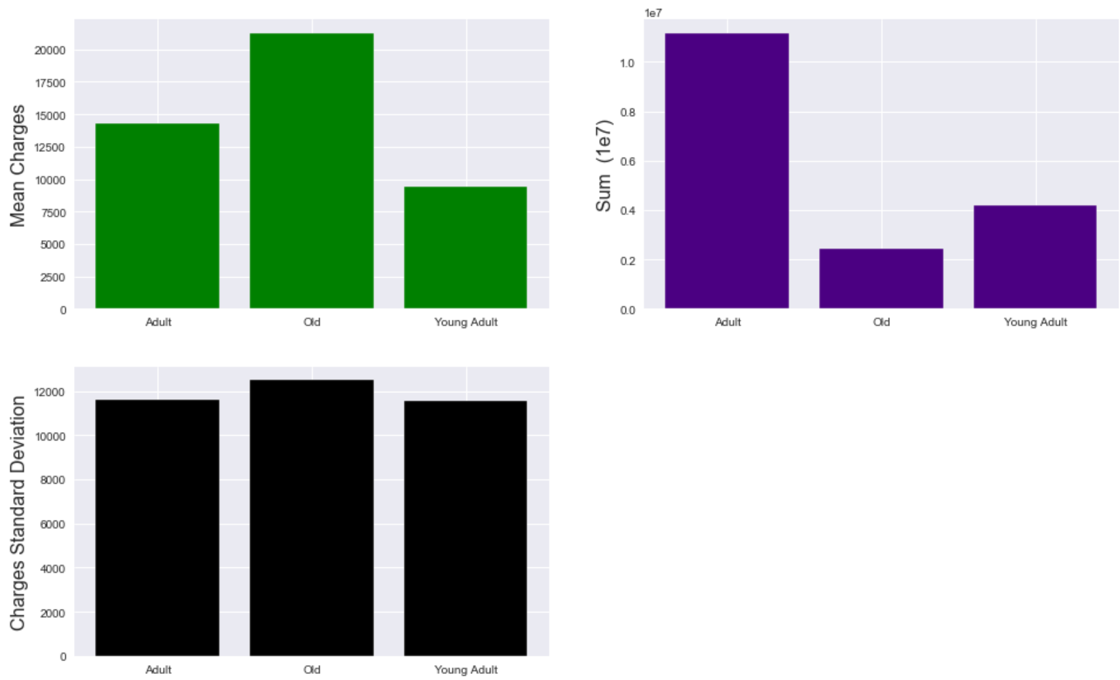
The graph below depicts that the mean cost per adult patient is less than $15,000 with a standard deviation of 12000. a mean of around less than 10000 and standard deviation of around 10000 for young adults and the mean cost for old age is the highest which shoots above 20,000 with a standard deviation of 13,000

This makes sense old age have highest avg cost. The older you get the more you spend.

[31]:
```
charge_avg_age=raw_data.groupby("age_cat")["charges"].mean()
labels_avg=charge_avg_age.keys()
charge_avg_age=charge_avg_age.tolist()

charge_sum_age=raw_data.groupby(["age_cat"])["charges"].sum()
```

```
labels_sum=charge_sum_age.keys()
charge_sum_age=charge_sum_age.tolist()

charge_std_age=raw_data.groupby(["age_cat"])["charges"].std()
labels_std=charge_std_age.keys()
charge_std_age=charge_std_age.tolist()


style.use("seaborn")
plt.figure(figsize=(16,10))
plt.subplot(2,2,1)
plt.bar(labels_avg, charge_avg_age, color="green")
plt.ylabel("Mean Charges", fontsize=16)
plt.subplot(2,2,2)
plt.bar(labels_sum, charge_sum_age, color="indigo")
plt.ylabel("Sum  (1e7)", fontsize=16)
plt.subplot(2,2,3)
plt.bar(labels_sum, charge_std_age, color="black")
plt.ylabel("Charges Standard Deviation", fontsize=16)
```

[31]: Text(0, 0.5, 'Charges Standard Deviation')



3.9 Remove Outliers

The histogram in the EDA section showed outliers in charges. Lets remove them.

[32]:
```
raw_data["log_charges"]=np.log(raw_data["charges"])
raw_data
```

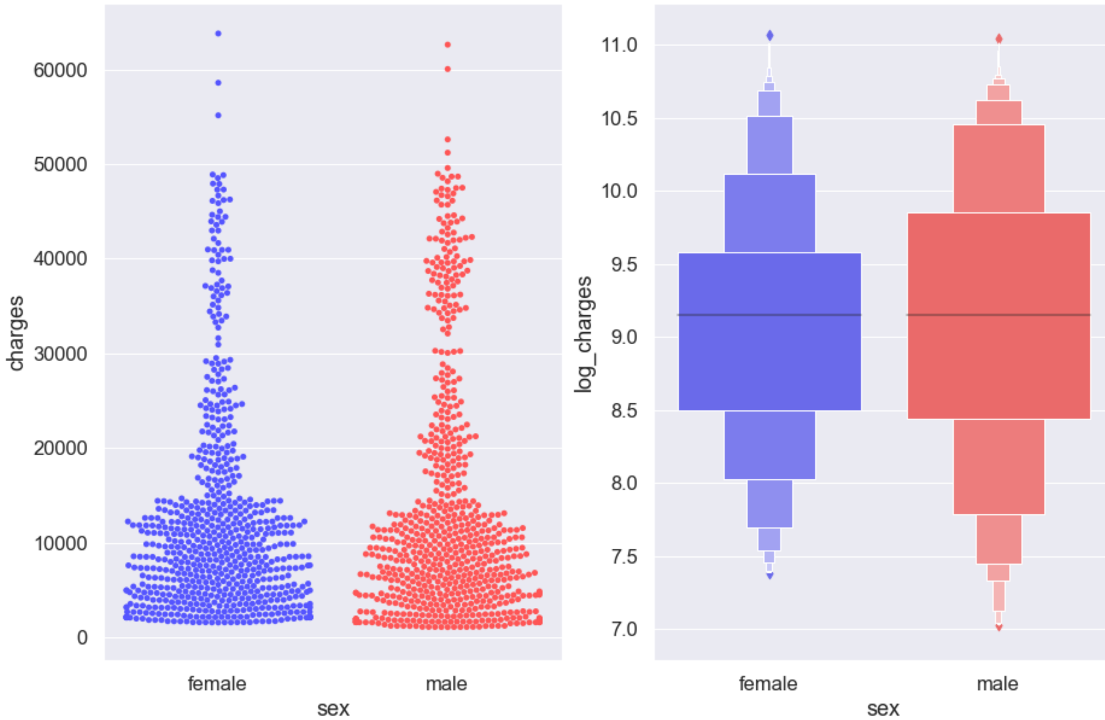
[32]:
```
      age     sex     bmi  children smoker     region      charges  \
0      19  female  27.900         0    yes  southwest  16884.92400
1      18    male  33.770         1     no  southeast   1725.55230
2      28    male  33.000         3     no  southeast   4449.46200
3      33    male  22.705         0     no  northwest  21984.47061
4      32    male  28.880         0     no  northwest   3866.85520
…     …      …       …        …      …         …          …
1333   50    male  30.970         3     no  northwest  10600.54830
1334   18  female  31.920         0     no  northeast   2205.98080
1335   18  female  36.850         0     no  southeast   1629.83350
1336   21  female  25.800         0     no  southwest   2007.94500
1337   61  female  29.070         0    yes  northwest  29141.36030

          age_cat  log_charges
0     Young Adult     9.734176
1     Young Adult     7.453302
2     Young Adult     8.400538
3           Adult     9.998092
4           Adult     8.260197
…             …           …
1333        Adult     9.268661
1334  Young Adult     7.698927
1335  Young Adult     7.396233
1336  Young Adult     7.604867
1337          Old    10.279914

[1338 rows x 9 columns]
```

3.10 Comparison health insurance charge and Gender

Visual below shows independency of charges on gender. With the mean lying around $10,000.

[34]:
```
plt.figure(figsize=(15,10))
sns.set(font_scale=1.5)
plt.subplot(1,2,1)
sns.swarmplot(raw_data["sex"], raw_data["charges"], palette ="seismic")
plt.subplot(1,2,2)
sns.boxenplot(raw_data["sex"], raw_data["log_charges"], palette ="seismic")
```

[34]: <matplotlib.axes._subplots.AxesSubplot at 0x7fccc0837350>

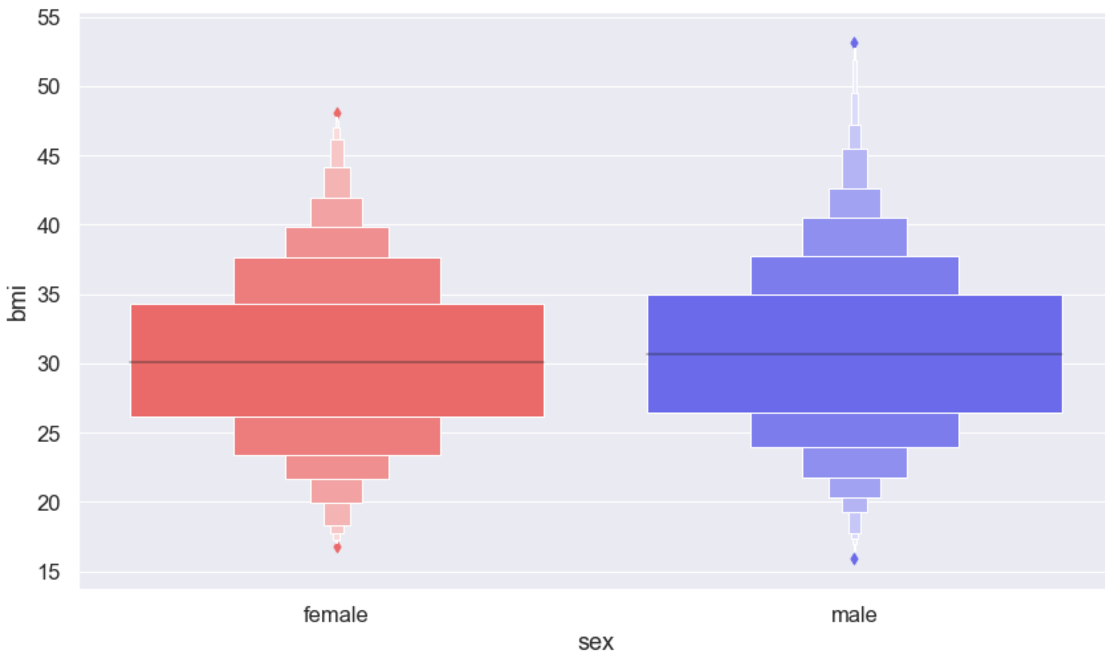


3.11 Comparison BMI charge and Gender

The distribution of BMI has a mean of around of 30 with upper quartile ranges from 34 to 35 and lower quartile ranges from 25 for both the gender.

[35]:
```
plt.figure(figsize=(14,8))
sns.set(font_scale=1.5)
sns.boxenplot(raw_data["sex"], raw_data["bmi"], palette ="seismic_r")
```

[35]: <matplotlib.axes._subplots.AxesSubplot at 0x7fccc156a290>



3.12 Feature Engineering

Lets add BMI catagory as catagorical value, under weight, normsl, overweight

[36]:
```
raw_data.loc[(raw_data.age<19), "bmi_cat"]="Underweight"
raw_data.loc[(raw_data.age>=19) & (raw_data.age<=25), "bmi_cat"]="Normal"
raw_data.loc[(raw_data.age>25) & (raw_data.age<=30), "bmi_cat"]="Overweight"
raw_data.loc[(raw_data.age>30), "bmi_cat"]="Obese"
raw_data
```

[36]:
```
      age     sex     bmi  children smoker     region      charges  \
0      19  female  27.900         0    yes  southwest  16884.92400
1      18    male  33.770         1     no  southeast   1725.55230
2      28    male  33.000         3     no  southeast   4449.46200
3      33    male  22.705         0     no  northwest  21984.47061
4      32    male  28.880         0     no  northwest   3866.85520
…     …       …     …       …      …         …          …
1333   50    male  30.970         3     no  northwest  10600.54830
1334   18  female  31.920         0     no  northeast   2205.98080
1335   18  female  36.850         0     no  southeast   1629.83350
1336   21  female  25.800         0     no  southwest   2007.94500
1337   61  female  29.070         0    yes  northwest  29141.36030

          age_cat  log_charges      bmi_cat
0     Young Adult     9.734176       Normal
1     Young Adult     7.453302  Underweight
```

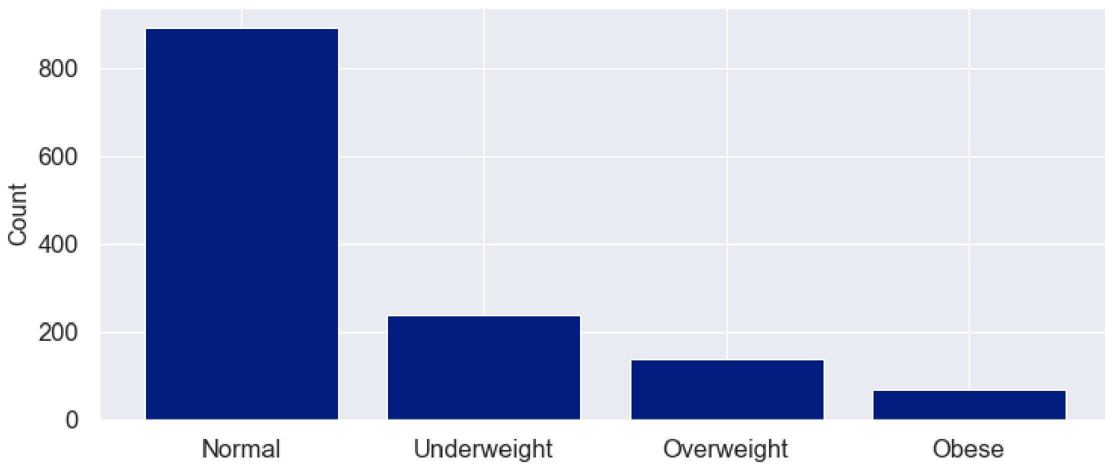
```
2     Young Adult     8.400538   Overweight
3           Adult     9.998092        Obese
4           Adult     8.260197        Obese
…           …          …          …
1333        Adult     9.268661        Obese
1334  Young Adult     7.698927  Underweight
1335  Young Adult     7.396233  Underweight
1336  Young Adult     7.604867       Normal
1337          Old    10.279914        Obese

[1338 rows x 10 columns]
```

3.13 Visualize BMI Distribution

A helpful visual that shows distribution. of BMI of over weight under weight, normal and obese. This dataset contains a lot of people who have normal BMI.

[37]:
```
bmi_val=raw_data["bmi_cat"].value_counts()
bmi_val=bmi_val.tolist()
style.use("seaborn-dark-palette")
labels=raw_data["bmi_cat"].unique()
plt.figure(figsize=(12,5))
plt.bar(labels, bmi_val)
plt.ylabel("Count", fontsize=16)
```

[37]:
```
Text(0, 0.5, 'Count')
```

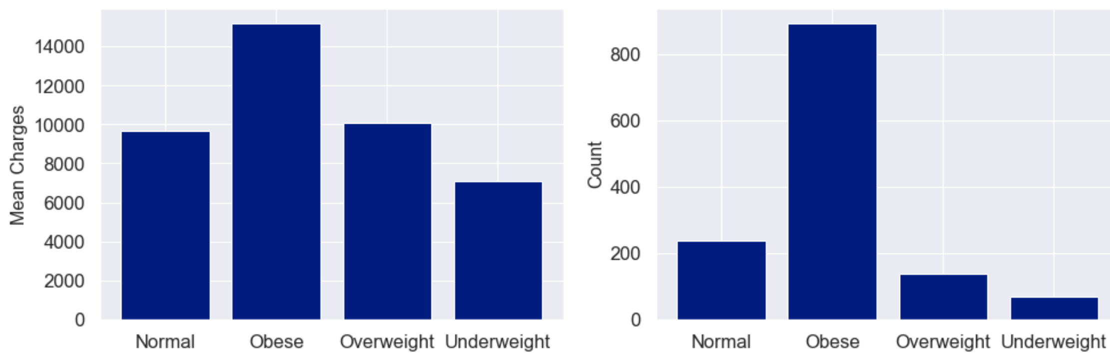


3.14 BMI Distrubution vs Charge

- we can see obesity has quite a impact on medical cost.Obese patients have a averagge cost above $14,000. Thus its better to keep our weights under control.

```
[38]: bmi_avg_charge=raw_data.groupby("bmi_cat")["charges"].mean()
labels_a=bmi_avg_charge.keys()
bmi_avg_charge=bmi_avg_charge.tolist()

bmi_count_charge=raw_data.groupby("bmi_cat")["charges"].count()
labels_c=bmi_count_charge.keys()
bmi_count_charge=bmi_count_charge.tolist()


style.use("seaborn-dark-palette")
plt.figure(figsize=(16,5))
plt.subplot(1,2,1)
plt.bar(labels_a, bmi_avg_charge)
plt.ylabel("Mean Charges", fontsize=16)

plt.subplot(1,2,2)
plt.bar(labels_c, bmi_count_charge)
plt.ylabel("Count", fontsize=16)
```

```
[38]: Text(0, 0.5, 'Count')
```

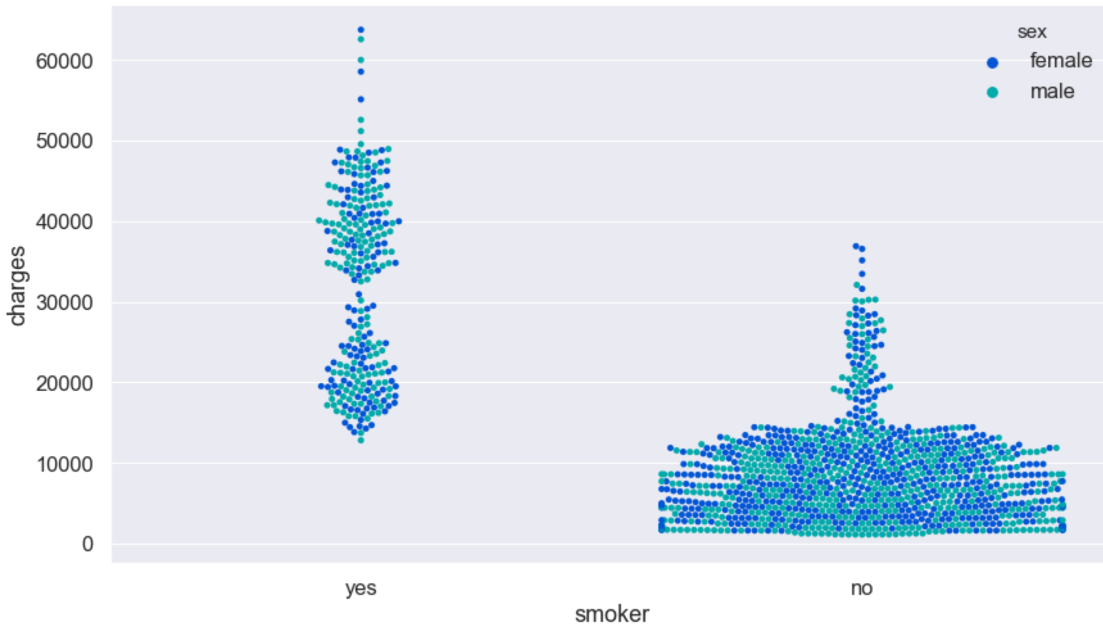


3.15 Relationship between Smoking and Charges

No surprise here. The more you smoke the higher your charge will be!

```
[39]: plt.figure(figsize=(14,8))
sns.set(font_scale=1.5)
sns.swarmplot(raw_data["smoker"], raw_data["charges"],hue=raw_data["sex"],␣
 ↪palette="winter")
```

```
[39]: <matplotlib.axes._subplots.AxesSubplot at 0x7fccc1b6cad0>
```



3.16 Standard Scalar

- BMI and Age range in tens where as Children range in once while Charges ranged in 5 digits. Thus to keep all on same page we use the standard scaler.

[41]:
```
std_scl=StandardScaler()
raw_data_std=std_scl.fit_transform(raw_data_c)
print("columns as age, bmi. children, charges")
print(raw_data_std)
```

```
columns as age, bmi. children, charges
[[-1.43876426 -0.45332    -0.90861367  0.2985838 ]
 [-1.50996545  0.5096211  -0.07876719 -0.95368917]
 [-0.79795355  0.38330685  1.58092576 -0.72867467]
 …
 [-1.50996545  1.0148781  -0.90861367 -0.96159623]
 [-1.29636188 -0.79781341 -0.90861367 -0.93036151]
 [ 1.55168573 -0.26138796 -0.90861367  1.31105347]]
```

[42]:
```
bmi_charg_c=raw_data_std[:,[1,3]]
print(bmi_charg_c)
print(bmi_charg_c.shape)
```

```
[[-0.45332     0.2985838 ]
 [ 0.5096211  -0.95368917]
 [ 0.38330685 -0.72867467]
 …
```

```
 [ 1.0148781  -0.96159623]
 [-0.79781341 -0.93036151]
 [-0.26138796  1.31105347]]
(1338, 2)
```

4 KMeans Cluster

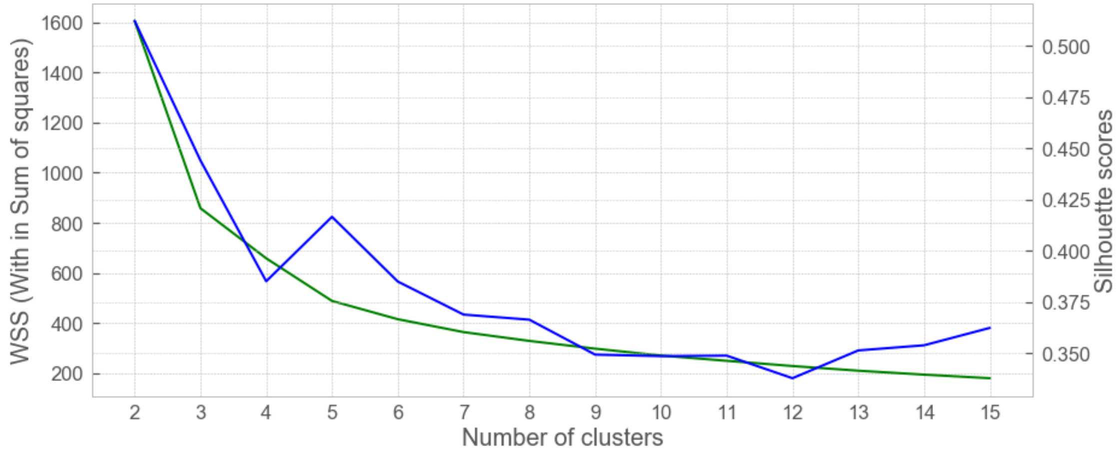
- To find the best number of cluster (n_clusters=k) we compute the WSS (Within sum of squares) Elbow method and Silhoutte scores for each "k".

[43]:
```
wss=[]
sil=[]
for k in range(2,16):
    kmeans=KMeans(n_clusters=k, random_state=1).fit(bmi_charg_c)
    wss.append(kmeans.inertia_)
    labels=kmeans.labels_
    silhoutte=silhouette_score(bmi_charg_c, labels, metric = 'euclidean')
    sil.append(silhoutte)
```

4.1 KMeans Cluster Visual

- From the plot we see the "elbow" at 3 and silhouutee score almost best at that point.

[45]:
```
k=range(2,16)
style.use("bmh")
fig,ax=plt.subplots(figsize=(14,6))
ax.set_facecolor("white")
ax.plot(k, wss, color="green")
ax.xaxis.set_major_locator(MaxNLocator(nbins=15, integer=True))
ax.set_xlabel("Number of clusters", fontsize=20)
ax.set_ylabel("WSS (With in Sum of squares)", fontsize=20)
ax2=ax.twinx()
ax2.plot(k, sil, color="blue")
ax2.set_ylabel("Silhouette scores", fontsize=20)
ax2.grid(True,color="silver")
plt.show()
```



4.2 KMeans Cluster ID

Adding numerical value to each row in which cluster they fall in

[46]:
```
k=3
kmeans=KMeans(n_clusters=k, random_state=1).fit(bmi_charg_c)
clusters=kmeans.labels_
centrids=kmeans.cluster_centers_
raw_data["clusters"]=clusters
raw_data
```

[46]:
```
      age     sex     bmi  children smoker     region      charges  \
0      19  female  27.900         0    yes  southwest  16884.92400
1      18    male  33.770         1     no  southeast   1725.55230
2      28    male  33.000         3     no  southeast   4449.46200
3      33    male  22.705         0     no  northwest  21984.47061
4      32    male  28.880         0     no  northwest   3866.85520
…     …     …       …       …      …          …            …
1333   50    male  30.970         3     no  northwest  10600.54830
1334   18  female  31.920         0     no  northeast   2205.98080
1335   18  female  36.850         0     no  southeast   1629.83350
1336   21  female  25.800         0     no  southwest   2007.94500
1337   61  female  29.070         0    yes  northwest  29141.36030

          age_cat  log_charges      bmi_cat  clusters
0     Young Adult     9.734176       Normal         1
1     Young Adult     7.453302  Underweight         0
2     Young Adult     8.400538   Overweight         0
3           Adult     9.998092        Obese         1
4           Adult     8.260197        Obese         1
…               …            …            …         …
```

```
1333         Adult     9.268661        Obese         0
1334  Young Adult     7.698927  Underweight         0
1335  Young Adult     7.396233  Underweight         0
1336  Young Adult     7.604867       Normal         1
1337           Old    10.279914        Obese         2

[1338 rows x 11 columns]
```

[47]:
```
raw_data2=raw_data.sort_values(["clusters"]).copy()
```

[48]:
```
for i in range(0,k+1):
    raw_data2["clusters"]=raw_data2["clusters"].replace(i, chr(i+65))

raw_data2
```

[48]:
```
      age     sex    bmi  children smoker     region      charges  \
945    56  female  35.80         1     no  southwest  11674.13000
449    35    male  38.60         1     no  southwest   4762.32900
895    61  female  44.00         0     no  southwest  13063.88300
894    62    male  32.11         0     no  northeast  13555.00490
1217   29    male  37.29         2     no  southeast   4058.11610
…     …      …      …          …    …        …           …
803    18  female  42.24         0    yes  southeast  38792.68560
770    61    male  36.10         3     no  southwest  27941.28758
759    18    male  38.17         0    yes  southeast  36307.79830
615    47  female  36.63         1    yes  southeast  42969.85270
1337   61  female  29.07         0    yes  northwest  29141.36030

          age_cat  log_charges      bmi_cat clusters
945         Adult     9.365131        Obese        A
449         Adult     8.468492        Obese        A
895           Old     9.477607        Obese        A
894           Old     9.514511        Obese        A
1217  Young Adult     8.308474   Overweight        A
…             …          …          …      …
803   Young Adult    10.565987  Underweight        C
770           Old    10.237861        Obese        C
759   Young Adult    10.499788  Underweight        C
615         Adult    10.668254        Obese        C
1337          Old    10.279914        Obese        C

[1338 rows x 11 columns]
```

[50]:
```
x=raw_data2.iloc[:,[2,6]].values
print(x.shape)
y=kmeans.fit_predict(x)
print(y.shape)
```

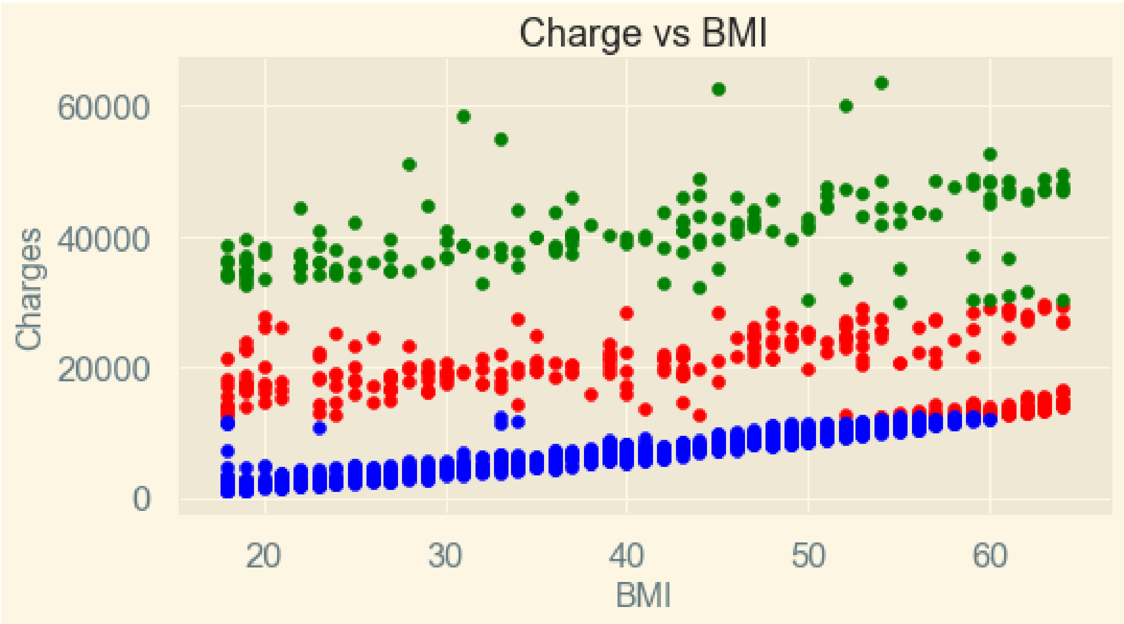
```
(1338, 2)
(1338,)
```

4.3 Does BMI impact charge?

- we have defined we got 3 distinct clusters. With BMI (15 to 35) has a expense of 10,000 to $30,000 where as higher BMI's have much higher cost.

[69]:
```
plt.figure(figsize=(8,4))
style.use("Solarize_Light2")
plt.scatter(x[y==0,0], x[y==0,1], color="red", label="A")
plt.scatter(x[y==1,0], x[y==1,1], color="blue", label="B")
plt.scatter(x[y==2,0], x[y==2,1], color="green", label="C")

plt.xlabel("BMI", fontsize=16)
plt.ylabel("Charges", fontsize=16)
plt.title("Charge vs BMI", fontsize=18)
```

[69]:
```
Text(0.5, 1.0, 'Charge vs BMI')
```



5 KMeans Cluster: Age

- We also Run the same clustering for "Age"

[52]:
```
age_charg_c=raw_data_std[:,[0,3]]
print(age_charg_c)
print(age_charg_c.shape)
```

```
[[-1.43876426  0.2985838 ]
 [-1.50996545 -0.95368917]
 [-0.79795355 -0.72867467]
 …
 [-1.50996545 -0.96159623]
 [-1.29636188 -0.93036151]
 [ 1.55168573  1.31105347]]
(1338, 2)
```

[53]:
```
wss=[]
sil=[]
for k in range(2,16):
    kmeans=KMeans(n_clusters=k, random_state=1).fit(age_charg_c)
    wss.append(kmeans.inertia_)
    labels=kmeans.labels_
    silhoutte=silhouette_score(age_charg_c, labels, metric = 'euclidean')
    sil.append(silhoutte)
```

5.1 KMeans Cluster Visual

- From the plot we see the "elbow" at 3 and silhoutee score almost best at that point.

[54]:
```
k=range(2,16)
style.use("bmh")
fig,ax=plt.subplots(figsize=(14,6))
ax.set_facecolor("white")
ax.plot(k, wss, color="green")
ax.xaxis.set_major_locator(MaxNLocator(nbins=15, integer=True))
ax.set_xlabel("No of clusters", fontsize=20)
ax.set_ylabel("WSS (With in Sum of squares)", fontsize=20)
ax2=ax.twinx()
ax2.plot(k, sil, color="blue")
ax2.set_ylabel("Silhouette scores", fontsize=20)
ax2.grid(True,color="silver")
plt.show()
```

[55]:
```
k=3
kmeans=KMeans(n_clusters=k, random_state=1).fit(age_charg_c)
clusters=kmeans.labels_
centrids=kmeans.cluster_centers_
raw_data["clusters"]=clusters
raw_data
```

[55]:

	age	sex	bmi	children	smoker	region	charges	\
0	19	female	27.900	0	yes	southwest	16884.92400	
1	18	male	33.770	1	no	southeast	1725.55230	
2	28	male	33.000	3	no	southeast	4449.46200	
3	33	male	22.705	0	no	northwest	21984.47061	
4	32	male	28.880	0	no	northwest	3866.85520	
…	…	…	…	…	…	…	…	
1333	50	male	30.970	3	no	northwest	10600.54830	
1334	18	female	31.920	0	no	northeast	2205.98080	
1335	18	female	36.850	0	no	southeast	1629.83350	
1336	21	female	25.800	0	no	southwest	2007.94500	
1337	61	female	29.070	0	yes	northwest	29141.36030	

	age_cat	log_charges	bmi_cat	clusters
0	Young Adult	9.734176	Normal	1
1	Young Adult	7.453302	Underweight	1
2	Young Adult	8.400538	Overweight	1
3	Adult	9.998092	Obese	1
4	Adult	8.260197	Obese	1
…	…	…	…	…
1333	Adult	9.268661	Obese	2
1334	Young Adult	7.698927	Underweight	1
1335	Young Adult	7.396233	Underweight	1
1336	Young Adult	7.604867	Normal	1

```
1337          Old    10.279914        Obese         2

[1338 rows x 11 columns]
```

[56]:
```python
raw_data2=raw_data.sort_values(["clusters"]).copy()
```

5.2 KMeans Cluster ID

[57]:
```python
for i in range(0,k+1):
    raw_data2["clusters"]=raw_data2["clusters"].replace(i, chr(i+65))

raw_data2
```

[57]:
```
      age     sex     bmi  children smoker     region      charges  \
668    62    male  32.015         0    yes  northeast  45710.20785
223    19    male  34.800         0    yes  southwest  34779.61500
1001   24    male  32.700         0    yes  southwest  34472.84100
987    45  female  27.645         1     no  northwest  28340.18885
240    23  female  36.670         2    yes  northeast  38511.62830
…     …       …       …         …      …          …
846    51  female  34.200         1     no  southwest   9872.70100
341    62    male  30.020         0     no  northwest  13352.09980
849    55    male  32.775         0     no  northwest  10601.63225
344    49  female  41.470         4     no  southeast  10977.20630
1337   61  female  29.070         0    yes  northwest  29141.36030

          age_cat  log_charges bmi_cat clusters
668           Old    10.730077   Obese        A
223   Young Adult    10.456787  Normal        A
1001  Young Adult    10.447927  Normal        A
987         Adult    10.252036   Obese        A
240   Young Adult    10.558716  Normal        A
…             …          …      …
846         Adult     9.197529   Obese        C
341           Old     9.499429   Obese        C
849         Adult     9.268763   Obese        C
344         Adult     9.303576   Obese        C
1337          Old    10.279914   Obese        C

[1338 rows x 11 columns]
```

[58]:
```python
x=raw_data2.iloc[:,[0,6]].values
print(x.shape)
y=kmeans.fit_predict(x)
print(y.shape)
```

```
(1338, 2)
```

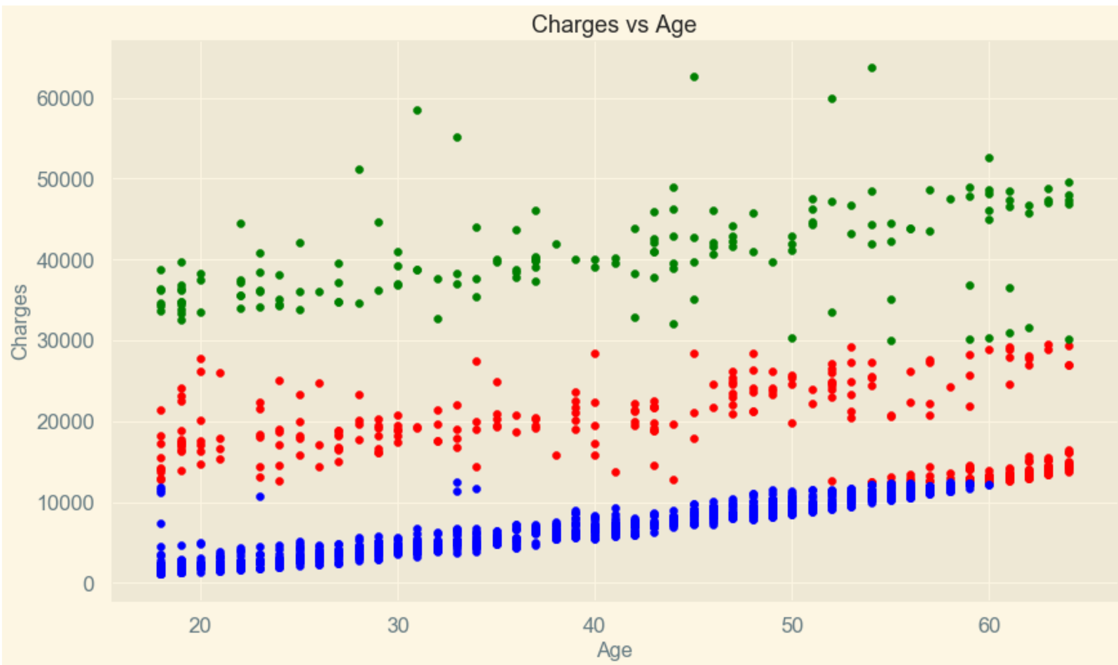
```
(1338,)
```

5.3 Charge vs Age

- We dont see much distinction about groups here with quite high overlaps. All the three expenses ranges has all the age groups

[70]:
```
plt.figure(figsize=(14,8))
style.use("Solarize_Light2")
plt.scatter(x[y==0,0], x[y==0,1], color="red", label="A")
plt.scatter(x[y==1,0], x[y==1,1], color="blue", label="B")
plt.scatter(x[y==2,0], x[y==2,1], color="green", label="C")

plt.xlabel("Age", fontsize=16)
plt.ylabel("Charges", fontsize=16)
plt.title("Charges vs Age", fontsize=18)
```

[70]:
```
Text(0.5, 1.0, 'Charges vs Age')
```



6 Hypothesis Testing

- We convert categorical variable "Smoker" as 0 and 1 or a continuous binary variable and run a OLS test. We also make our hypothesis.
- H0 - Charges are independent of variables
- H1- Chrges are dependent on variables

[60]:
```
raw_data2["smoker"]=raw_data2["smoker"].replace(["yes", "no"],[1,0])
pval=ols("charges~bmi+age+children+smoker", data=raw_data).fit()
```

7 Conclusions

[61]:
```
print(pval.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                charges   R-squared:                       0.750
Model:                            OLS   Adj. R-squared:                  0.749
Method:                 Least Squares   F-statistic:                     998.1
Date:                Sun, 15 Nov 2020   Prob (F-statistic):               0.00
Time:                        13:33:56   Log-Likelihood:                -13551.
No. Observations:                1338   AIC:                         2.711e+04
Df Residuals:                    1333   BIC:                         2.714e+04
Df Model:                           4
Covariance Type:            nonrobust
================================================================================
=
                    coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
-
Intercept      -1.21e+04    941.984    -12.848      0.000    -1.4e+04
-1.03e+04
smoker[T.yes]  2.381e+04    411.220     57.904      0.000     2.3e+04
2.46e+04
bmi             321.8514     27.378     11.756      0.000     268.143
375.559
age             257.8495     11.896     21.675      0.000     234.512
281.187
children        473.5023    137.792      3.436      0.001     203.190
743.814
==============================================================================
Omnibus:                      301.480   Durbin-Watson:                   2.087
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              722.157
Skew:                           1.215   Prob(JB):                    1.53e-157
Kurtosis:                       5.654   Cond. No.                         292.
==============================================================================

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

7.1 Interpretation

- all the 4 independent variable has a Pvalue of less than 0.05 thus we reject the null hypothesis. and conclude that "Charges" are dependent on the mentioned variables.

Individuals both female, male and of all ages should keep their BMI at a healthy level, they should not smoke, and should be aware that more children may lead to an increase in health insurance.

7.2 Limitations-Further

I would have liked to use a linear regression model as well to see if we could make a prediction of future charges based on the data we were given. I do not believe there were any limitations, because it has been reported by CDC that charges have been increasing drastically, and this supports that and also states the reasons as to why it is happening. Ultimately, aging is apart of human process so it is likely to increase in everyone.