DM Project

Devansh Gupta

C037 Section B batch b2


```
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px

import warnings
warnings.filterwarnings('ignore')
```

1) Choosing a data set

```
df = pd.read_csv("/Users/devansh/Documents/ML
Projects/project4/diabetes.csv")
df.head()
```

```
   Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin
BMI  \
0            6      148             72             35        0  33.6

1            1       85             66             29        0  26.6

2            8      183             64              0        0  23.3

3            1       89             66             23       94  28.1

4            0      137             40             35      168  43.1


   DiabetesPedigreeFunction  Age  Outcome
0                     0.627   50        1
1                     0.351   31        0
2                     0.672   32        1
3                     0.167   21        0
4                     2.288   33        1
```

2) Reading the data and understanding the dataset

```
df.dtypes
```

```
Pregnancies                   int64
Glucose                       int64
BloodPressure                 int64
SkinThickness                 int64
Insulin                       int64
BMI                         float64
DiabetesPedigreeFunction    float64
Age                           int64
Outcome                       int64
dtype: object
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 9 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   Pregnancies               768 non-null    int64  
 1   Glucose                   768 non-null    int64  
 2   BloodPressure             768 non-null    int64  
 3   SkinThickness             768 non-null    int64  
 4   Insulin                   768 non-null    int64  
 5   BMI                       768 non-null    float64
 6   DiabetesPedigreeFunction  768 non-null    float64
 7   Age                       768 non-null    int64  
 8   Outcome                   768 non-null    int64  
dtypes: float64(2), int64(7)
memory usage: 54.1 KB
```

```
df.describe().round(2).style.background_gradient()
```

```
<pandas.io.formats.style.Styler at 0x7ff324d584c0>
```

```
df.isnull().sum()
```

```
Pregnancies                 0
Glucose                     0
BloodPressure               0
SkinThickness               0
Insulin                     0
BMI                         0
DiabetesPedigreeFunction    0
Age                         0
Outcome                     0
dtype: int64
```

3) Cleaning the dataset

```
df[['Glucose','BloodPressure','SkinThickness','Insulin','BMI','Diabete
sPedigreeFunction','Age']] =
```

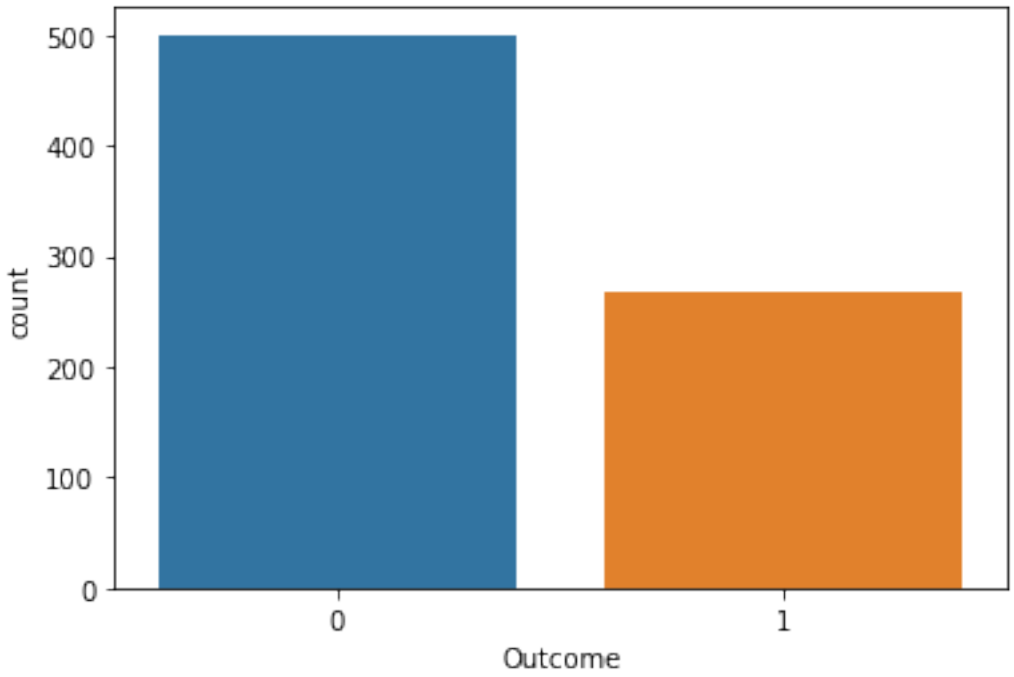
```
df[['Glucose','BloodPressure','SkinThickness','Insulin','BMI','Diabete
sPedigreeFunction','Age']].replace(0,np.NaN)
df.fillna(df.mean(), inplace = True) #Filled Mising values with Mean
df.isnull().sum()

#all we did here is replaced 0 will nul and replaced all the null by
the mean of the column
```

```
Pregnancies                 0
Glucose                     0
BloodPressure               0
SkinThickness               0
Insulin                     0
BMI                         0
DiabetesPedigreeFunction    0
Age                         0
Outcome                     0
dtype: int64
```

4) Visulaztion of dataset

```
sns.countplot(df.Outcome)
```

```
<AxesSubplot:xlabel='Outcome', ylabel='count'>
```




```
lis=["don't have diabetes","have diabetes"]
have_or_not = df["Outcome"].value_counts().tolist()
values = [have_or_not[0], have_or_not[1]]
fig = px.pie(values=df['Outcome'].value_counts(), names=lis ,
```

```
width=800, height=400, color_discrete_sequence=["skyblue","black"]
              ,title="percentage between have diabetes or not")
fig.show()
```


```
{"data":[{"hovertemplate":"label=%{label}<br>value=%{value}<extra></
extra>","values":[500,268],"domain":{"y":[0,1],"x":[0,1]},"labels":
["don't have diabetes","have
diabetes"],"showlegend":true,"name":"","legendgroup":"","type":"pie"}]
,"config":{"plotlyServerURL":"https://plot.ly"},"layout":
{"height":400,"width":800,"title":{"text":"percentage between have
diabetes or not"},"legend":{"tracegroupgap":0},"piecolorway":
["skyblue","black"],"template":{"data":{"contourcarpet":[{"colorbar":
{"outlinewidth":0,"ticks":""},"type":"contourcarpet"}],"scattermapbox"
:[{"type":"scattermapbox","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"mesh3d":[{"colorbar":
{"outlinewidth":0,"ticks":""},"type":"mesh3d"}],"heatmap":
[{"colorbar":{"outlinewidth":0,"ticks":""},"colorscale":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"heatmap"}],"pie":
[{"automargin":true,"type":"pie"}],"carpet":[{"aaxis":
{"linecolor":"white","minorgridcolor":"white","endlinecolor":"#2a3f5f"
,"startlinecolor":"#2a3f5f","gridcolor":"white"},"baxis":
{"linecolor":"white","minorgridcolor":"white","endlinecolor":"#2a3f5f"
,"startlinecolor":"#2a3f5f","gridcolor":"white"},"type":"carpet"}],"ba
r":[{"error_x":{"color":"#2a3f5f"},"error_y":
{"color":"#2a3f5f"},"type":"bar","marker":{"line":
{"width":0.5,"color":"#E5ECF6"},"pattern":
{"solidity":0.2,"fillmode":"overlay","size":10}}}],"barpolar":
[{"type":"barpolar","marker":{"line":
{"width":0.5,"color":"#E5ECF6"},"pattern":
{"solidity":0.2,"fillmode":"overlay","size":10}}}],"scatter3d":
[{"line":{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"scatter3d","marker":
{"colorbar":{"outlinewidth":0,"ticks":""}}}],"contour":[{"colorbar":
{"outlinewidth":0,"ticks":""},"colorscale":[[0,"#0d0887"],
[0.1111111111111111,"#46039f"],[0.2222222222222222,"#7201a8"],
[0.3333333333333333,"#9c179e"],[0.4444444444444444,"#bd3786"],
[0.5555555555555556,"#d8576b"],[0.6666666666666666,"#ed7953"],
[0.7777777777777778,"#fb9f3a"],[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"contour"}],"histogram2d":[{"colorbar":
{"outlinewidth":0,"ticks":""},"colorscale":[[0,"#0d0887"],
[0.1111111111111111,"#46039f"],[0.2222222222222222,"#7201a8"],
[0.3333333333333333,"#9c179e"],[0.4444444444444444,"#bd3786"],
[0.5555555555555556,"#d8576b"],[0.6666666666666666,"#ed7953"],
[0.7777777777777778,"#fb9f3a"],[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"histogram2d"}],"scatterpolar":
```

```
[{"type":"scatterpolar","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"histogram":
[{"type":"histogram","marker":{"pattern":
{"solidity":0.2,"fillmode":"overlay","size":10}}}],"histogram2dcontour
":[{"colorbar":{"outlinewidth":0,"ticks":""},"colorscale":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"histogram2dcontour"}],"parcoords":[{"line":
{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"parcoords"}],"scatterpolargl":
[{"type":"scatterpolargl","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"heatmapgl":[{"colorbar":
{"outlinewidth":0,"ticks":""},"colorscale":[[0,"#0d0887"],
[0.1111111111111111,"#46039f"],[0.2222222222222222,"#7201a8"],
[0.3333333333333333,"#9c179e"],[0.4444444444444444,"#bd3786"],
[0.5555555555555556,"#d8576b"],[0.6666666666666666,"#ed7953"],
[0.7777777777777778,"#fb9f3a"],[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"heatmapgl"}],"scattercarpet":
[{"type":"scattercarpet","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"choropleth":[{"colorbar":
{"outlinewidth":0,"ticks":""},"type":"choropleth"}],"scatterternary":
[{"type":"scatterternary","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"scatter":
[{"type":"scatter","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"table":[{"cells":{"fill":
{"color":"#EBF0F8"},"line":{"color":"white"}},"header":{"fill":
{"color":"#C8D4E3"},"line":
{"color":"white"}},"type":"table"}],"scattergeo":
[{"type":"scattergeo","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"surface":[{"colorbar":
{"outlinewidth":0,"ticks":""},"colorscale":[[0,"#0d0887"],
[0.1111111111111111,"#46039f"],[0.2222222222222222,"#7201a8"],
[0.3333333333333333,"#9c179e"],[0.4444444444444444,"#bd3786"],
[0.5555555555555556,"#d8576b"],[0.6666666666666666,"#ed7953"],
[0.7777777777777778,"#fb9f3a"],[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"surface"}],"scattergl":
[{"type":"scattergl","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}]},"layout":{"ternary":{"aaxis":
{"linecolor":"white","ticks":"","gridcolor":"white"},"baxis":
{"linecolor":"white","ticks":"","gridcolor":"white"},"caxis":
{"linecolor":"white","ticks":"","gridcolor":"white"},"bgcolor":"#E5ECF
6"},"autotypenumbers":"strict","shapedefaults":{"line":
{"color":"#2a3f5f"}},"annotationdefaults":
{"arrowwidth":1,"arrowcolor":"#2a3f5f","arrowhead":0},"coloraxis":
{"colorbar":{"outlinewidth":0,"ticks":""}},"title":{"x":5.0e-
2},"hoverlabel":{"align":"left"},"colorscale":{"diverging":
[[0,"#8e0152"],[0.1,"#c51b7d"],[0.2,"#de77ae"],[0.3,"#f1b6da"],
```

```
[0.4,"#fde0ef"],[0.5,"#f7f7f7"],[0.6,"#e6f5d0"],[0.7,"#b8e186"],
[0.8,"#7fbc41"],[0.9,"#4d9221"],[1,"#276419"]],"sequentialminus":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],[1,"#f0f921"]],"sequential":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]]},"hovermode":"closest","mapbox":
{"style":"light"},"paper_bgcolor":"white","scene":{"zaxis":
{"linecolor":"white","showbackground":true,"zerolinecolor":"white","gr
idwidth":2,"ticks":"","backgroundcolor":"#E5ECF6","gridcolor":"white"}
,"xaxis":
{"linecolor":"white","showbackground":true,"zerolinecolor":"white","gr
idwidth":2,"ticks":"","backgroundcolor":"#E5ECF6","gridcolor":"white"}
,"yaxis":
{"linecolor":"white","showbackground":true,"zerolinecolor":"white","gr
idwidth":2,"ticks":"","backgroundcolor":"#E5ECF6","gridcolor":"white"}
},"font":{"color":"#2a3f5f"},"xaxis":{"linecolor":"white","title":
{"standoff":15},"zerolinewidth":2,"automargin":true,"zerolinecolor":"w
hite","ticks":"","gridcolor":"white"},"polar":{"angularaxis":
{"linecolor":"white","ticks":"","gridcolor":"white"},"radialaxis":
{"linecolor":"white","ticks":"","gridcolor":"white"},"bgcolor":"#E5ECF
6"},"plot_bgcolor":"#E5ECF6","geo":
{"subunitcolor":"white","lakecolor":"white","landcolor":"#E5ECF6","sho
wland":true,"showlakes":true,"bgcolor":"white"},"yaxis":
{"linecolor":"white","title":
{"standoff":15},"zerolinewidth":2,"automargin":true,"zerolinecolor":"w
hite","ticks":"","gridcolor":"white"},"colorway":
["#636efa","#EF553B","#00cc96","#ab63fa","#FFA15A","#19d3f3","#FF6692"
,"#B6E880","#FF97FF","#FECB52"]}}}}
```


```
fig = px.bar(df['Age'].value_counts(), height=400, width = 700)
fig.show()
```

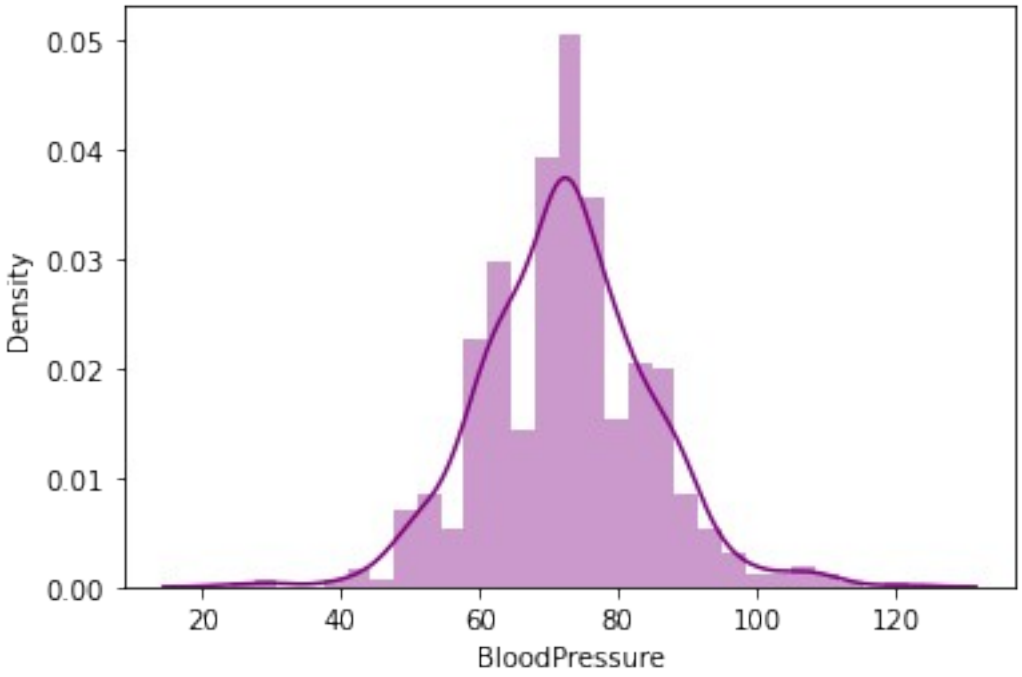

```
{"data":[{"y":
[72,63,48,46,38,35,33,32,29,24,22,21,19,18,17,16,16,16,15,14,13,13,13,
12,10,8,8,8,8,7,6,6,5,5,5,5,5,4,4,4,4,3,3,3,3,2,2,1,1,1,1,1],"hovertem
plate":"variable=Age<br>index=%{x}<br>value=%{y}<extra></
extra>","alignmentgroup":"True","x":
[22,21,25,24,23,28,26,27,29,31,41,30,37,42,33,38,36,32,45,34,46,43,40,
39,35,50,51,52,44,58,47,54,49,48,57,53,60,66,63,62,55,67,56,59,65,69,6
1,72,81,64,70,68],"showlegend":true,"offsetgroup":"Age","name":"Age","
textposition":"auto","legendgroup":"Age","orientation":"v","xaxis":"x"
,"type":"bar","marker":{"color":"#636efa","pattern":
{"shape":""}},"yaxis":"y"}],"config":{"plotlyServerURL":"https://
plot.ly"},"layout":
```

```
{"height":400,"width":700,"barmode":"relative","legend":{"title":
{"text":"variable"},"tracegroupgap":0},"margin":{"t":60},"xaxis":
{"anchor":"y","title":{"text":"index"},"domain":[0,1]},"template":
{"data":{"contourcarpet":[{"colorbar":
{"outlinewidth":0,"ticks":""},"type":"contourcarpet"}],"scattermapbox"
:[{"type":"scattermapbox","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"mesh3d":[{"colorbar":
{"outlinewidth":0,"ticks":""},"type":"mesh3d"}],"heatmap":
[{"colorbar":{"outlinewidth":0,"ticks":""},"colorscale":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"heatmap"}],"pie":
[{"automargin":true,"type":"pie"}],"carpet":[{"aaxis":
{"linecolor":"white","minorgridcolor":"white","endlinecolor":"#2a3f5f"
,"startlinecolor":"#2a3f5f","gridcolor":"white"},"baxis":
{"linecolor":"white","minorgridcolor":"white","endlinecolor":"#2a3f5f"
,"startlinecolor":"#2a3f5f","gridcolor":"white"},"type":"carpet"}],"ba
r":[{"error_x":{"color":"#2a3f5f"},"error_y":
{"color":"#2a3f5f"},"type":"bar","marker":{"line":
{"width":0.5,"color":"#E5ECF6"},"pattern":
{"solidity":0.2,"fillmode":"overlay","size":10}}}],"barpolar":
[{"type":"barpolar","marker":{"line":
{"width":0.5,"color":"#E5ECF6"},"pattern":
{"solidity":0.2,"fillmode":"overlay","size":10}}}],"scatter3d":
[{"line":{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"scatter3d","marker":
{"colorbar":{"outlinewidth":0,"ticks":""}}}],"contour":[{"colorbar":
{"outlinewidth":0,"ticks":""},"colorscale":[[0,"#0d0887"],
[0.1111111111111111,"#46039f"],[0.2222222222222222,"#7201a8"],
[0.3333333333333333,"#9c179e"],[0.4444444444444444,"#bd3786"],
[0.5555555555555556,"#d8576b"],[0.6666666666666666,"#ed7953"],
[0.7777777777777778,"#fb9f3a"],[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"contour"}],"histogram2d":[{"colorbar":
{"outlinewidth":0,"ticks":""},"colorscale":[[0,"#0d0887"],
[0.1111111111111111,"#46039f"],[0.2222222222222222,"#7201a8"],
[0.3333333333333333,"#9c179e"],[0.4444444444444444,"#bd3786"],
[0.5555555555555556,"#d8576b"],[0.6666666666666666,"#ed7953"],
[0.7777777777777778,"#fb9f3a"],[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"histogram2d"}],"scatterpolar":
[{"type":"scatterpolar","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"histogram":
[{"type":"histogram","marker":{"pattern":
{"solidity":0.2,"fillmode":"overlay","size":10}}}],"histogram2dcontour
":[{"colorbar":{"outlinewidth":0,"ticks":""},"colorscale":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
```

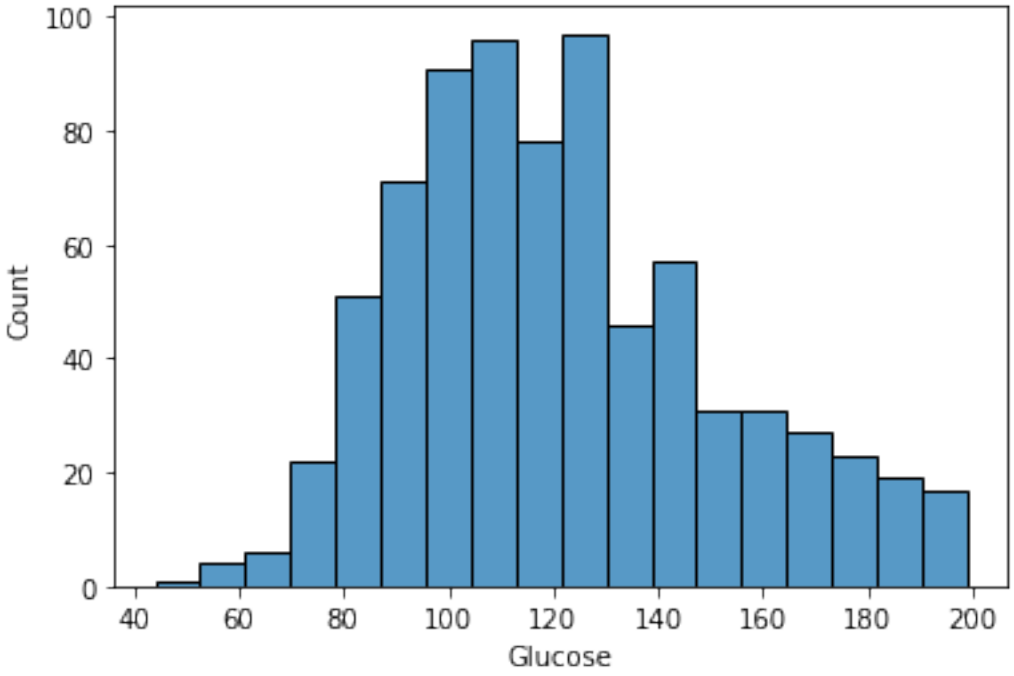
```
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"histogram2dcontour"}],"parcoords":[{"line":
{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"parcoords"}],"scatterpolargl":
[{"type":"scatterpolargl","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"heatmapgl":[{"colorbar":
{"outlinewidth":0,"ticks":""},"colorscale":[[0,"#0d0887"],
[0.1111111111111111,"#46039f"],[0.2222222222222222,"#7201a8"],
[0.3333333333333333,"#9c179e"],[0.4444444444444444,"#bd3786"],
[0.5555555555555556,"#d8576b"],[0.6666666666666666,"#ed7953"],
[0.7777777777777778,"#fb9f3a"],[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"heatmapgl"}],"scattercarpet":
[{"type":"scattercarpet","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"choropleth":[{"colorbar":
{"outlinewidth":0,"ticks":""},"type":"choropleth"}],"scatterternary":
[{"type":"scatterternary","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"scatter":
[{"type":"scatter","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"table":[{"cells":{"fill":
{"color":"#EBF0F8"},"line":{"color":"white"}},"header":{"fill":
{"color":"#C8D4E3"},"line":
{"color":"white"}},"type":"table"}],"scattergeo":
[{"type":"scattergeo","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}],"surface":[{"colorbar":
{"outlinewidth":0,"ticks":""},"colorscale":[[0,"#0d0887"],
[0.1111111111111111,"#46039f"],[0.2222222222222222,"#7201a8"],
[0.3333333333333333,"#9c179e"],[0.4444444444444444,"#bd3786"],
[0.5555555555555556,"#d8576b"],[0.6666666666666666,"#ed7953"],
[0.7777777777777778,"#fb9f3a"],[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"surface"}],"scattergl":
[{"type":"scattergl","marker":{"colorbar":
{"outlinewidth":0,"ticks":""}}}]},"layout":{"ternary":{"aaxis":
{"linecolor":"white","ticks":"","gridcolor":"white"},"baxis":
{"linecolor":"white","ticks":"","gridcolor":"white"},"caxis":
{"linecolor":"white","ticks":"","gridcolor":"white"},"bgcolor":"#E5ECF
6"},"autotypenumbers":"strict","shapedefaults":{"line":
{"color":"#2a3f5f"}},"annotationdefaults":
{"arrowwidth":1,"arrowcolor":"#2a3f5f","arrowhead":0},"coloraxis":
{"colorbar":{"outlinewidth":0,"ticks":""}},"title":{"x":5.0e-
2},"hoverlabel":{"align":"left"},"colorscale":{"diverging":
[[0,"#8e0152"],[0.1,"#c51b7d"],[0.2,"#de77ae"],[0.3,"#f1b6da"],
[0.4,"#fde0ef"],[0.5,"#f7f7f7"],[0.6,"#e6f5d0"],[0.7,"#b8e186"],
[0.8,"#7fbc41"],[0.9,"#4d9221"],[1,"#276419"]],"sequentialminus":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],[1,"#f0f921"]],"sequential":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
```

```
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]]},"hovermode":"closest","mapbox":
{"style":"light"},"paper_bgcolor":"white","scene":{"zaxis":
{"linecolor":"white","showbackground":true,"zerolinecolor":"white","gr
idwidth":2,"ticks":"","backgroundcolor":"#E5ECF6","gridcolor":"white"}
,"xaxis":
{"linecolor":"white","showbackground":true,"zerolinecolor":"white","gr
idwidth":2,"ticks":"","backgroundcolor":"#E5ECF6","gridcolor":"white"}
,"yaxis":
{"linecolor":"white","showbackground":true,"zerolinecolor":"white","gr
idwidth":2,"ticks":"","backgroundcolor":"#E5ECF6","gridcolor":"white"}
},"font":{"color":"#2a3f5f"},"xaxis":{"linecolor":"white","title":
{"standoff":15},"zerolinewidth":2,"automargin":true,"zerolinecolor":"w
hite","ticks":"","gridcolor":"white"},"polar":{"angularaxis":
{"linecolor":"white","ticks":"","gridcolor":"white"},"radialaxis":
{"linecolor":"white","ticks":"","gridcolor":"white"},"bgcolor":"#E5ECF
6"},"plot_bgcolor":"#E5ECF6","geo":
{"subunitcolor":"white","lakecolor":"white","landcolor":"#E5ECF6","sho
wland":true,"showlakes":true,"bgcolor":"white"},"yaxis":
{"linecolor":"white","title":
{"standoff":15},"zerolinewidth":2,"automargin":true,"zerolinecolor":"w
hite","ticks":"","gridcolor":"white"},"colorway":
["#636efa","#EF553B","#00cc96","#ab63fa","#FFA15A","#19d3f3","#FF6692"
,"#B6E880","#FF97FF","#FECB52"]}},"yaxis":{"anchor":"x","title":
{"text":"value"},"domain":[0,1]}}}
```

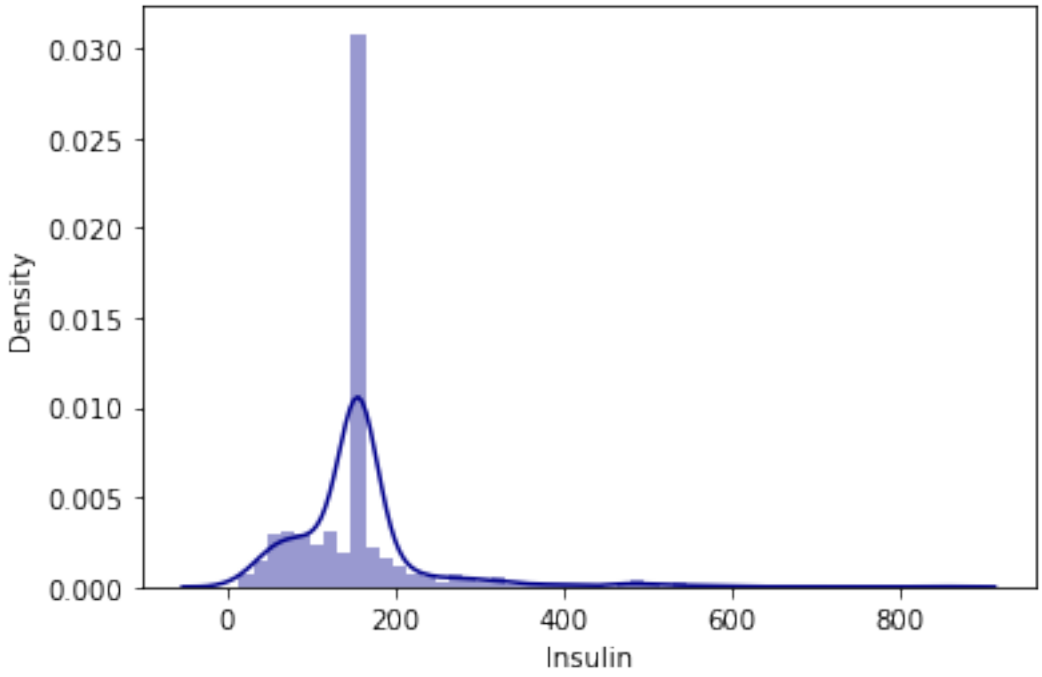

```
sns.distplot(df["BloodPressure"], color = "purple");
```



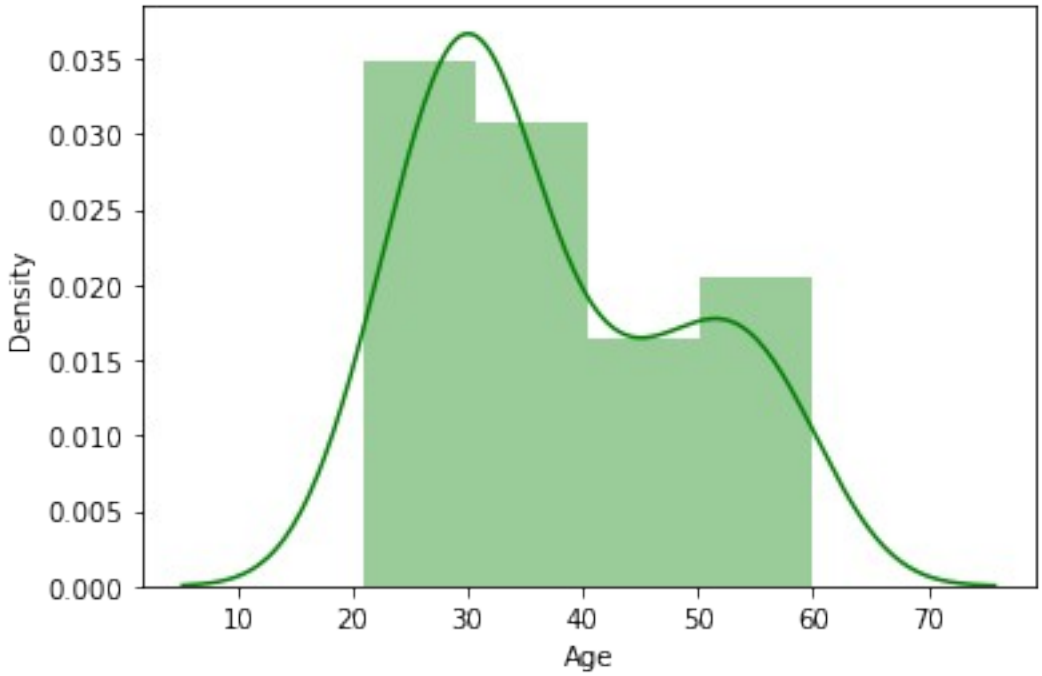

```
sns.histplot(df["Glucose"]);
```



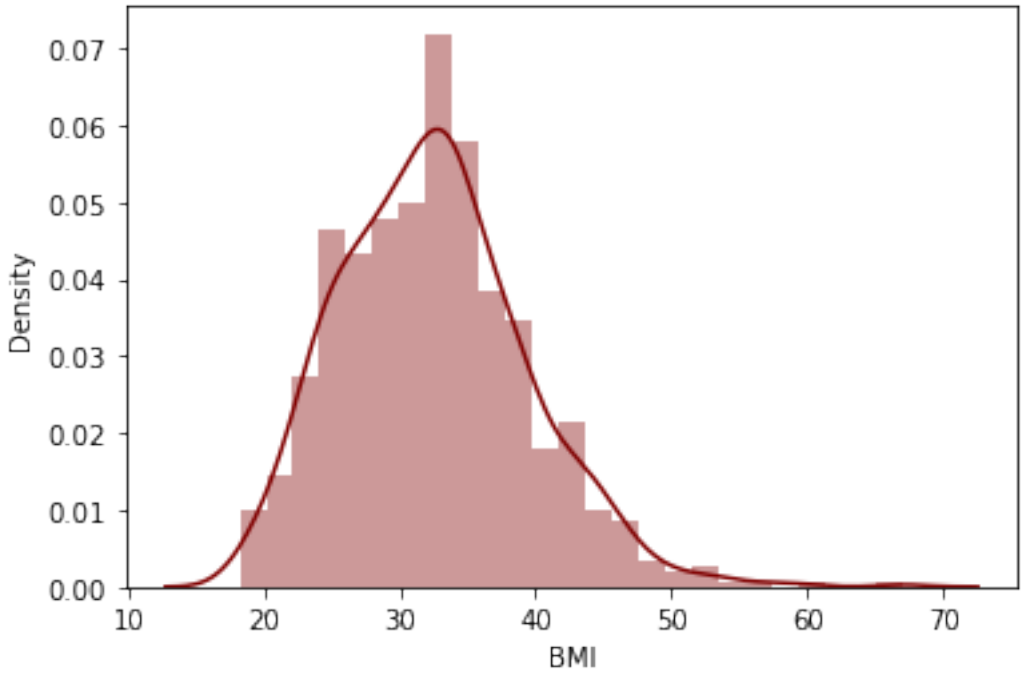

```
sns.distplot(df["Insulin"], color = "DarkBlue");
```




```
sns.distplot(df["Age"][:50], color = "Green");
```

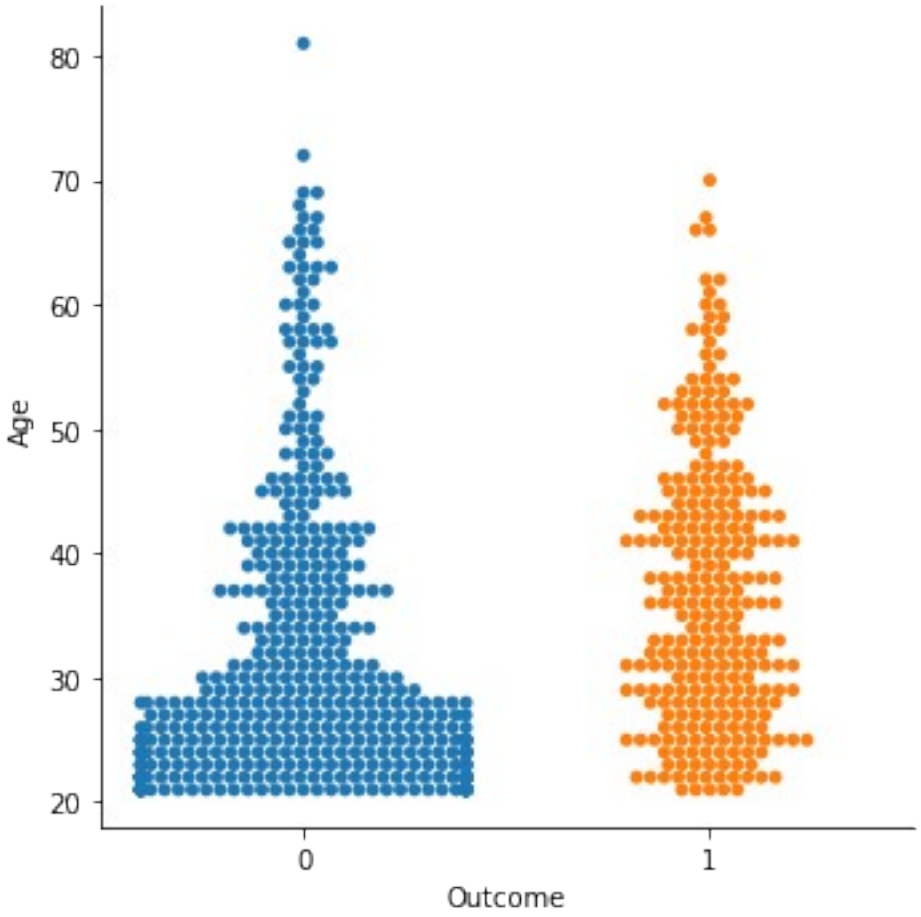



```
sns.distplot(df["BMI"], color = "Maroon");
```

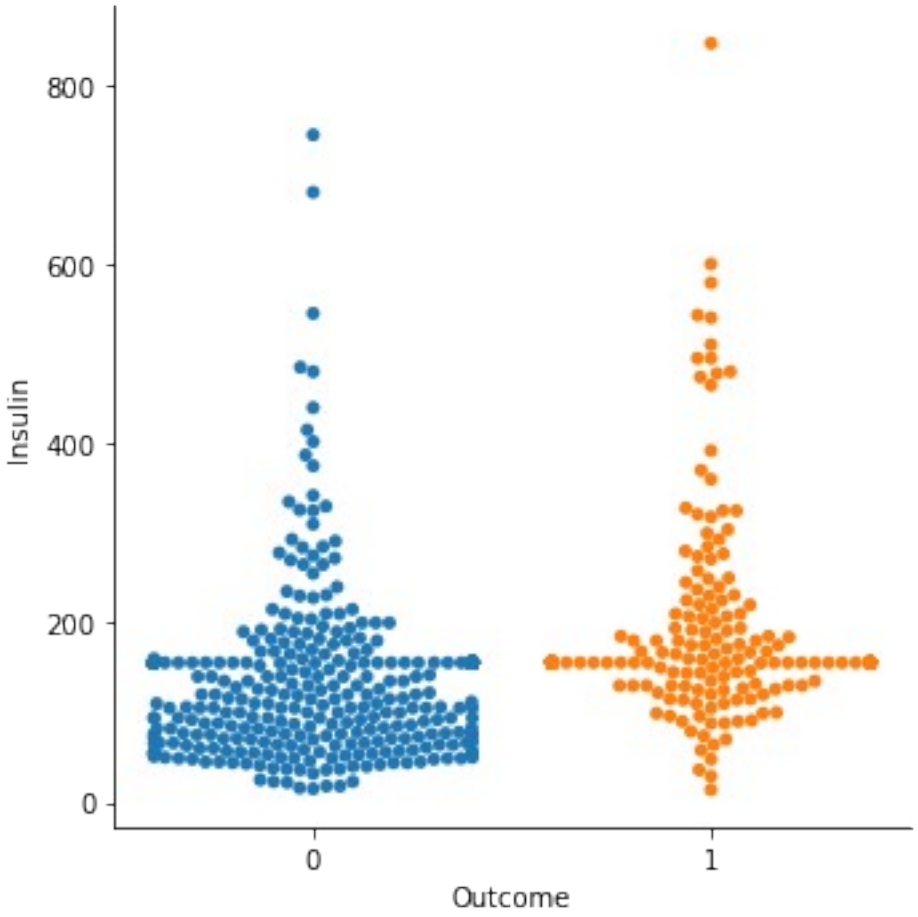


Advance cat plot

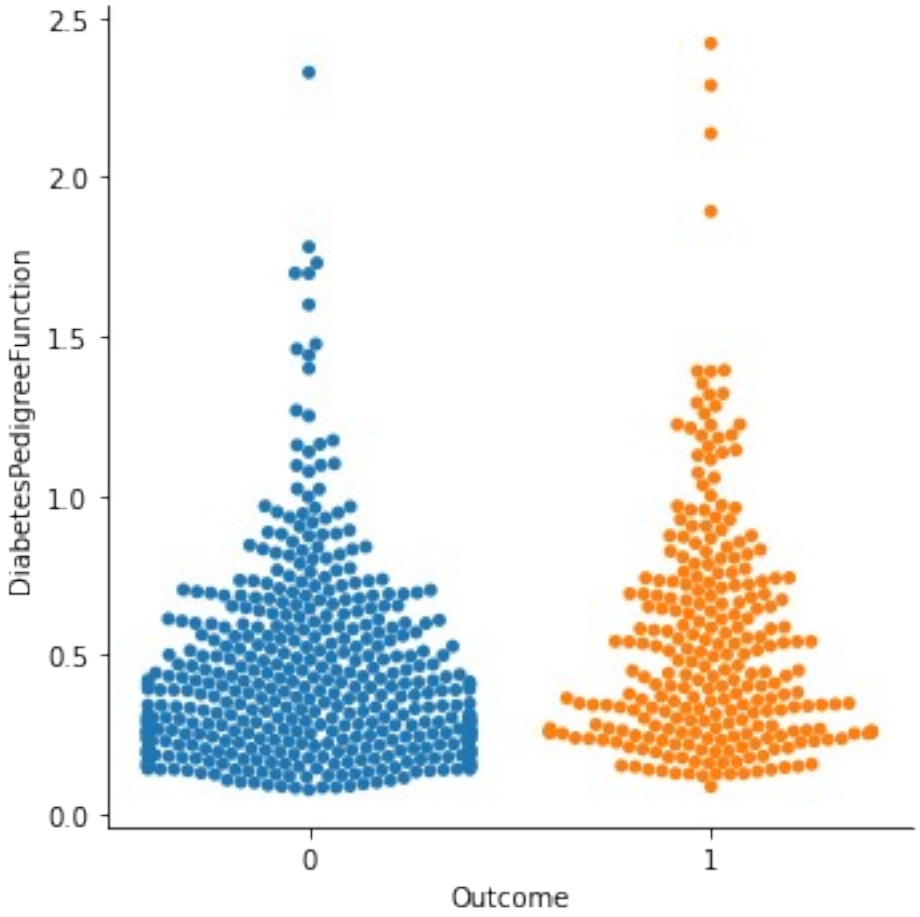
```
# cat plot
sns.catplot(x = "Outcome", y = "Age", hue = "Outcome", kind = "swarm",
data = df)
plt.show()
```



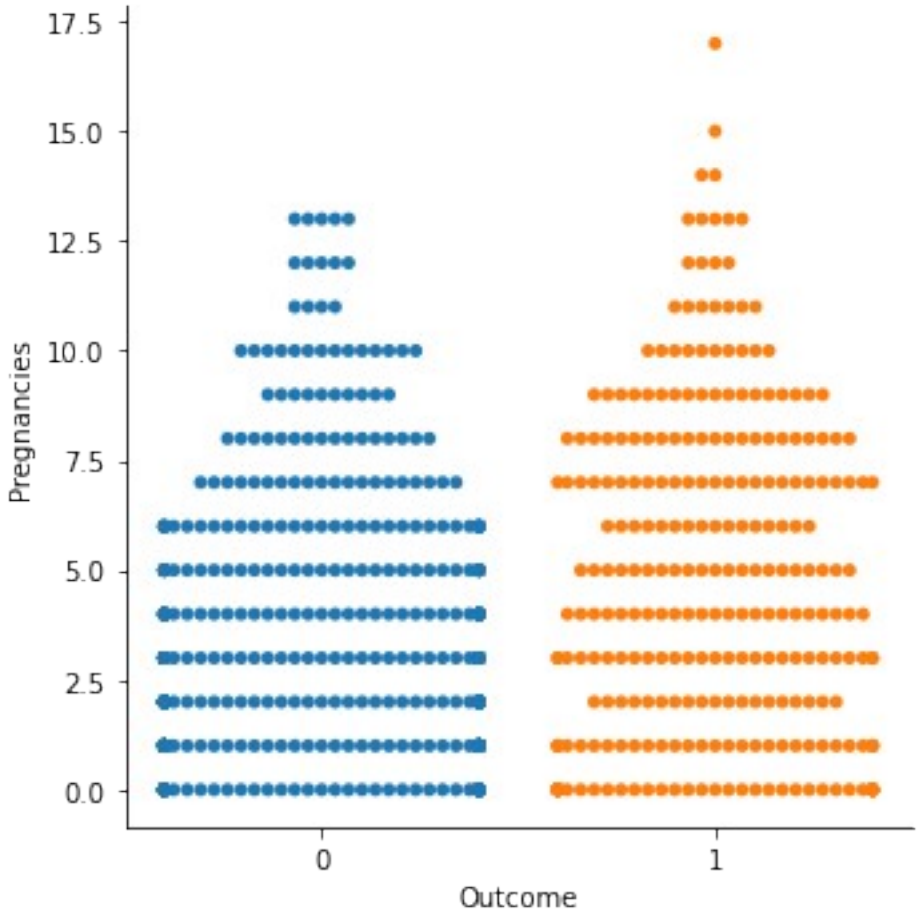

```
sns.catplot(x = "Outcome", y = "Insulin", hue = "Outcome", kind =
"swarm", data = df)
plt.show()
```




```
sns.catplot(x = "Outcome", y = "DiabetesPedigreeFunction", hue =
"Outcome", kind = "swarm", data = df)
plt.show()
```




```
sns.catplot(x = "Outcome", y = "Pregnancies", hue = "Outcome", kind =
"swarm", data = df)
plt.show()
```



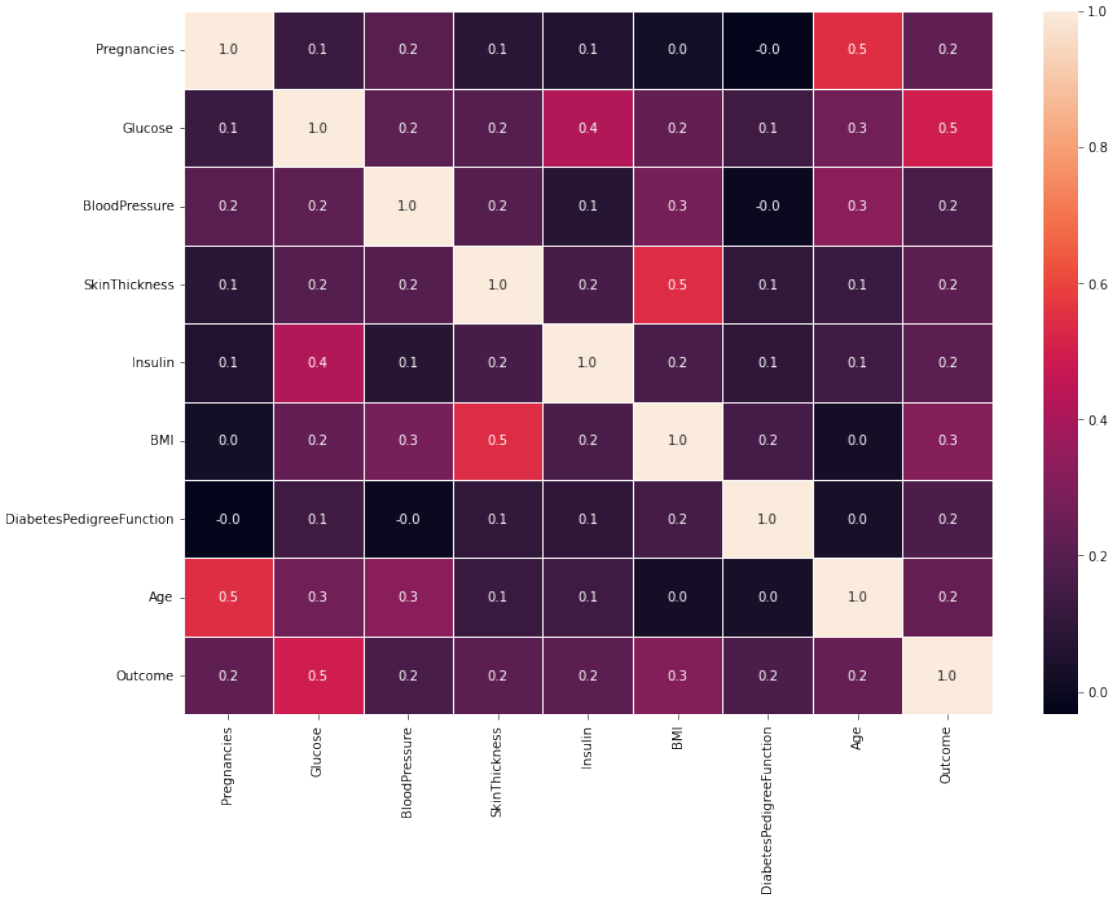
5)Feature Selection for making the output more and more effective

```
df.corr()
```

```
                          Pregnancies   Glucose  BloodPressure
SkinThickness  \
Pregnancies                  1.000000  0.127911       0.208522
0.082989
Glucose                      0.127911  1.000000       0.218367
0.192991
BloodPressure                0.208522  0.218367       1.000000
0.192816
SkinThickness                0.082989  0.192991       0.192816
1.000000
Insulin                      0.056027  0.420157       0.072517
0.158139
BMI                          0.021565  0.230941       0.281268
0.542398
DiabetesPedigreeFunction    -0.033523  0.137060      -0.002763
0.100966
Age                          0.544341  0.266534       0.324595
```

```
0.127872
Outcome                       0.221898  0.492928       0.166074
0.215299

                           Insulin       BMI  DiabetesPedigreeFunction
\
Pregnancies               0.056027  0.021565                 -0.033523

Glucose                   0.420157  0.230941                  0.137060

BloodPressure             0.072517  0.281268                 -0.002763

SkinThickness             0.158139  0.542398                  0.100966

Insulin                   1.000000  0.166586                  0.098634

BMI                       0.166586  1.000000                  0.153400

DiabetesPedigreeFunction  0.098634  0.153400                  1.000000

Age                       0.136734  0.025519                  0.033561

Outcome                   0.214411  0.311924                  0.173844


                               Age   Outcome
Pregnancies               0.544341  0.221898
Glucose                   0.266534  0.492928
BloodPressure             0.324595  0.166074
SkinThickness             0.127872  0.215299
Insulin                   0.136734  0.214411
BMI                       0.025519  0.311924
DiabetesPedigreeFunction  0.033561  0.173844
Age                       1.000000  0.238356
Outcome                   0.238356  1.000000
```

```
plt.figure(figsize = (14, 10))
sns.heatmap(df.corr(), annot = True, fmt = ".1f", linewidths = .7)
plt.show()
```




```
from sklearn.ensemble import RandomForestClassifier
clf = RandomForestClassifier()
x=df[df.columns[:8]]
y=df.Outcome
clf.fit(x,y)
feature_imp = pd.DataFrame(clf.feature_importances_,index=x.columns)
feature_imp.sort_values(by = 0 , ascending = False)
```

```
                                0
Glucose                  0.261190
BMI                      0.154962
Age                      0.133530
DiabetesPedigreeFunction 0.127818
Insulin                  0.085612
BloodPressure            0.081630
Pregnancies              0.079389
SkinThickness            0.075869
```

first 4 features displayed maybe important for us!!We might neglect the rest

```
from sklearn.model_selection import train_test_split
```

```
features = df[["Glucose",'BMI','Age','DiabetesPedigreeFunction']]
labels = df.Outcome
features.head()
```

```
   Glucose   BMI  Age  DiabetesPedigreeFunction
0    148.0  33.6   50                     0.627
1     85.0  26.6   31                     0.351
2    183.0  23.3   32                     0.672
3     89.0  28.1   21                     0.167
4    137.0  43.1   33                     2.288
```

```
features_train,features_test,labels_train,labels_test =
train_test_split(features,labels,stratify=df.Outcome,test_size=0.4)
```

Decision Trees

```
from sklearn.tree import DecisionTreeClassifier
dtclf = DecisionTreeClassifier()
dtclf.fit(features_train,labels_train)
dtclf.score(features_test,labels_test)
```

```
0.7045454545454546
```

SVM

```
from sklearn import svm
clf = svm.SVC(kernel="linear")
clf.fit(features_train,labels_train)
clf.score(features_test,labels_test)
```

```
0.7954545454545454
```

Naive Bayes

```
from sklearn import naive_bayes
nbclf = naive_bayes.GaussianNB()
nbclf.fit(features_train,labels_train)
nbclf.score(features_test,labels_test)
```

```
0.8051948051948052
```

K Neighbor

```
from sklearn.neighbors import KNeighborsClassifier
knnclf = KNeighborsClassifier(n_neighbors=2)
knnclf.fit(features_train,labels_train)
print(knnclf.score(features_test,labels_test))
```

```
0.7272727272727273
```

Logistic Regression

```
from sklearn.linear_model import LogisticRegression
clf1 = LogisticRegression()
clf1.fit(features_train,labels_train)
clf1.score(features_test,labels_test)
```

```
0.8051948051948052
```

Accurary Table

```
algos = ["Support Vector Machine","Decision Tree","Logistic
Regression","K Nearest Neighbor","Naive Bayes"]
clfs =
[svm.SVC(kernel="linear"),DecisionTreeClassifier(),LogisticRegression(
),KNeighborsClassifier(n_neighbors=2),naive_bayes.GaussianNB()]
result = []

for clff in clfs:
    clff.fit(features_train,labels_train)
    acc = clff.score(features_test,labels_test)
    result.append(acc)
result_df = pd.DataFrame(result,index=algos)
result_df.columns=["Accuracy"]
result_df.sort_values(by="Accuracy",ascending=False)
```

```
                       Accuracy
Logistic Regression    0.805195
Naive Bayes            0.805195
Support Vector Machine 0.795455
K Nearest Neighbor     0.727273
Decision Tree          0.711039
```

Cross Validation

```
#Cross Validation
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
kfold =KFold(n_splits=10)

algos = ["Support Vector Machine","Decision Tree","Logistic
Regression","K Nearest Neighbor","Naive Bayes"]
clfs =
[svm.SVC(kernel="linear"),DecisionTreeClassifier(),LogisticRegression(
),KNeighborsClassifier(n_neighbors=2),naive_bayes.GaussianNB()]
cv_results=[]
for classifiers in clfs:
    cv_score =
cross_val_score(classifiers,features,labels,cv=kfold,scoring="accuracy
")
```

```
    cv_results.append(cv_score.mean())
cv_mean = pd.DataFrame(cv_results,index=algos)
cv_mean.columns=["Accuracy"]
cv_mean.sort_values(by="Accuracy",ascending=False)
```

	Accuracy
Naive Bayes	0.776042
Logistic Regression	0.773394
Support Vector Machine	0.770796
K Nearest Neighbor	0.706955
Decision Tree	0.703025

Observations- We can see the accuracy changed a bit this time.It is because we have done cross validation and trained and tested the algorithms on diffrent combinations of data. From the above output, it is clear that for this dataset, SVM, Logistic Regression and Naive Bayes works better