Bversity

1st Ever Virtual College for Biotech

Name :Rajashree.U

Contact:rajashree5820@gmail.com

Mobile:+916383331696

Title: Python Exploratory Data analysis

**Answer the following questions below and upload them in the google form.**

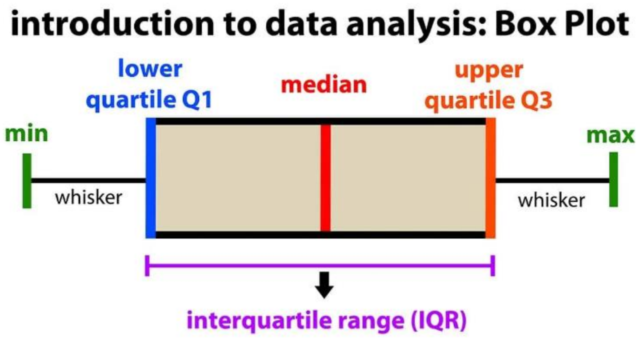
1. How does the distribution of feature “fractal_dimension_worst” differ between benign and malignant cases?
2. What is the range of values for the feature “radius_mean” and how skewed is its distribution?
3. Are there any outliers in feature “area_mean” and how might they affect analysis?
4. Based on the EDA, what factors seem to be most relevant to predicting breast cancer diagnosis?
5. What limitations are there in the data, and how might they affect our conclusions?

# PYTHON EXPLORATORY DATA ANALYSIS

Generating a suitable visualization can help examine the differences in the distribution of the features between malignant and benign tumours.

## Boxplot to examine the differences in the distribution of the features between the malignant and the benign tumours.

***Box Plot:*** A box plot gives information about the variability and dispersion of the data in the form of a box. The middle line represents the median value with the first and third quartiles(Center tendencies) on either end edges. Lines(Whisker) on either side of the box represent the Minimum and the Maximum.


introduction to data analysis: Box Plot
lower quartile Q1
median
upper quartile Q3
min
max
whisker
whisker
interquartile range (IQR)

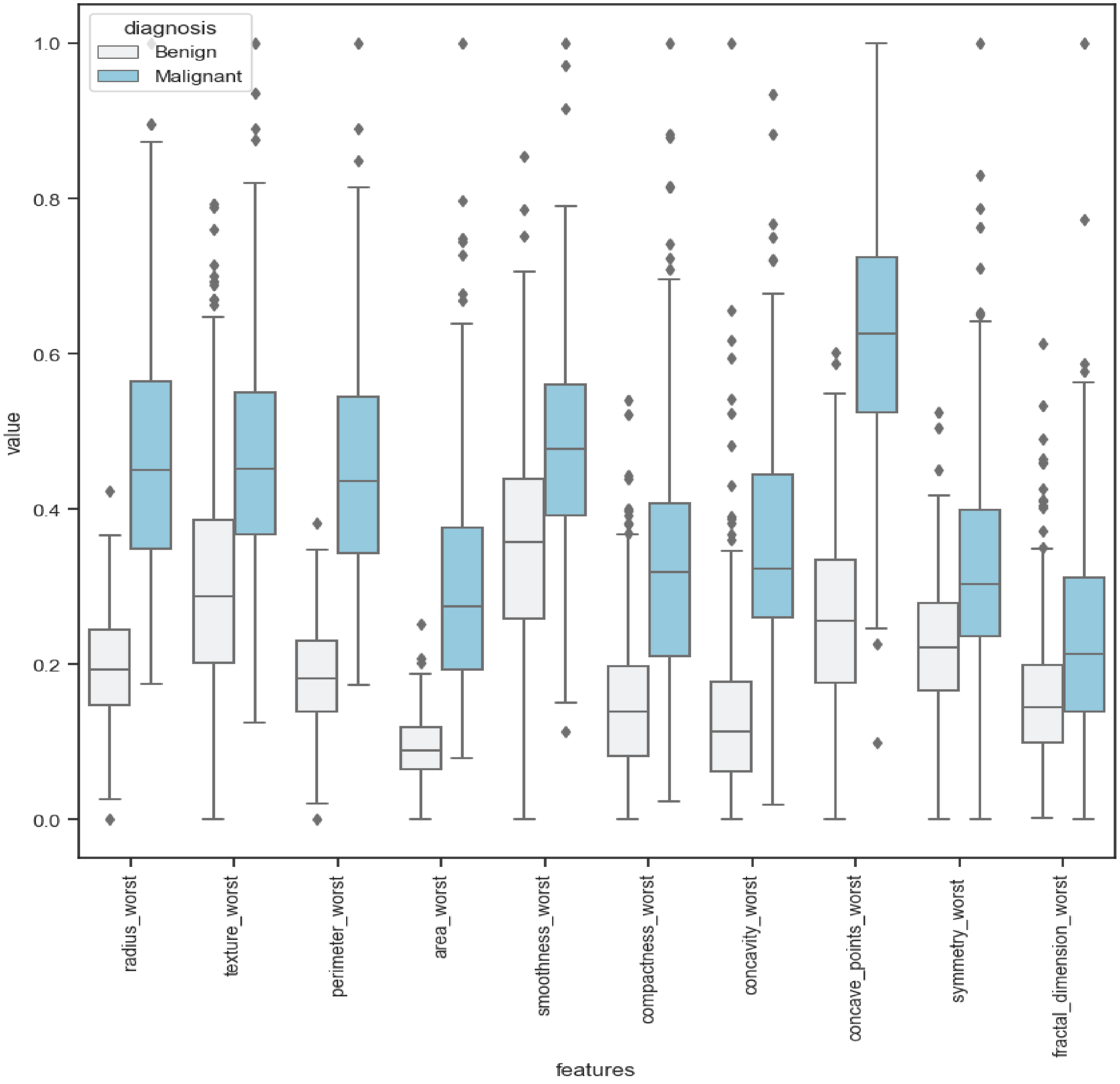
**#Using normalised data**

**Input**:

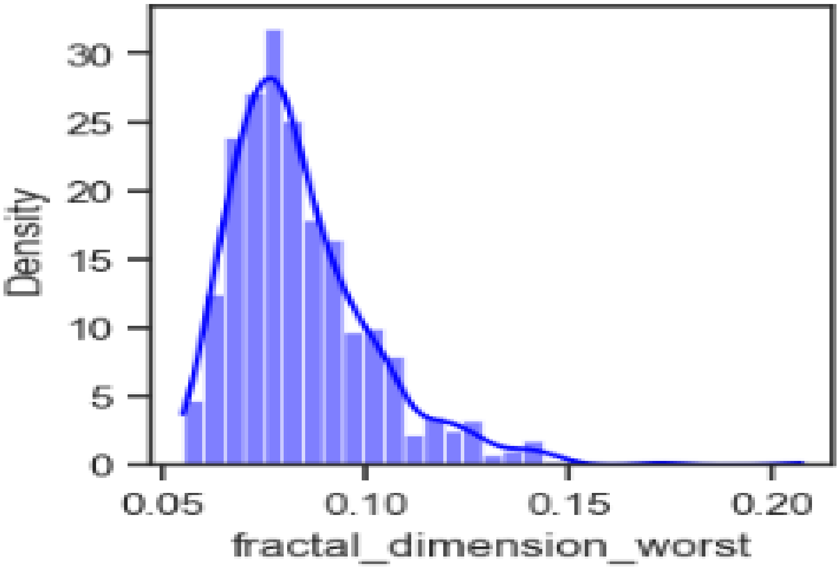
```
data = pd.concat([norm_df["diagnosis"],norm_df.iloc[:,21:31]], axis = 1)
data = pd.melt(data,id_vars="diagnosis",var_name="features",value_name='value')
plt.figure(figsize=(10,10))
g = sns.boxplot(x="features", y="value", hue="diagnosis", data=data , color = "skyblue")
label_0 = 'Benign'
g.legend_.texts[0].set_text(label_0)
label_1 = "Malignant"
g.legend_.texts[1].set_text(label_1)
plt.xticks(rotation=90)
```

**Output:**

```
(array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9]),
 [Text(0, 0, 'radius_worst'),
  Text(1, 0, 'texture_worst'),
  Text(2, 0, 'perimeter_worst'),
  Text(3, 0, 'area_worst'),
  Text(4, 0, 'smoothness_worst'),
  Text(5, 0, 'compactness_worst'),
  Text(6, 0, 'concavity_worst'),
  Text(7, 0, 'concave_points_worst'),
  Text(8, 0, 'symmetry_worst '),
  Text(9, 0, 'fractal_dimension_worst')]])
```



**Boxplot of certain features(fractal_dimension_worst)**



## Distribution plot of fractal_dimension_worst

## Observation:

1. Benign cases cluster around a lower value (approximately 0.35-0.4), while malignant cases cluster around a higher value (approximately 0.5-0.55). The benign and malignant cases appear to form two distinct clusters with different central tendencies.

2. Most benign cases have values below 0.45, while most malignant cases have values above 0.45. There is minimal overlap between the two distributions.

3. The benign distribution appears slightly right-skewed, with a few cases extending towards higher values. The malignant distribution also appears right-skewed but with a longer tail and potentially a secondary peak around 0.6-0.65.

4. Some benign cases exceed 0.5 and some malignant cases fall below 0.45. There are a few potential outliers in both groups

5. The distribution appears to be Gaussian.

## The range of values for the feature “radius_mean” and how skewed is its distribution

## #Getting the summary statistics of the area_mean feature using the describe

**Input:**

```
df["area_mean"].describe()
```

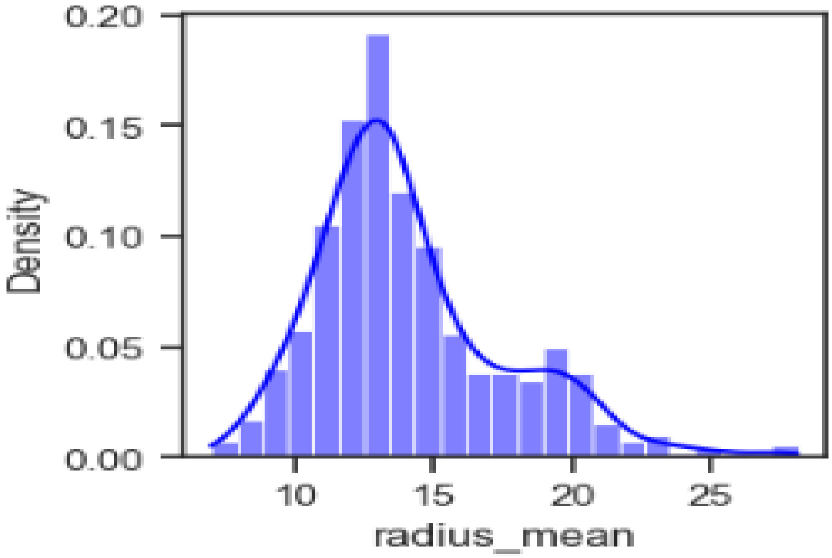
**Output:**

```
count    569.000000
mean      14.137253
std       3.443406
min       6.981000
25%       11.840000
50%       13.400000
75%       15.710000
max       28.110000
Name: radius_mean, dtype: float64
```

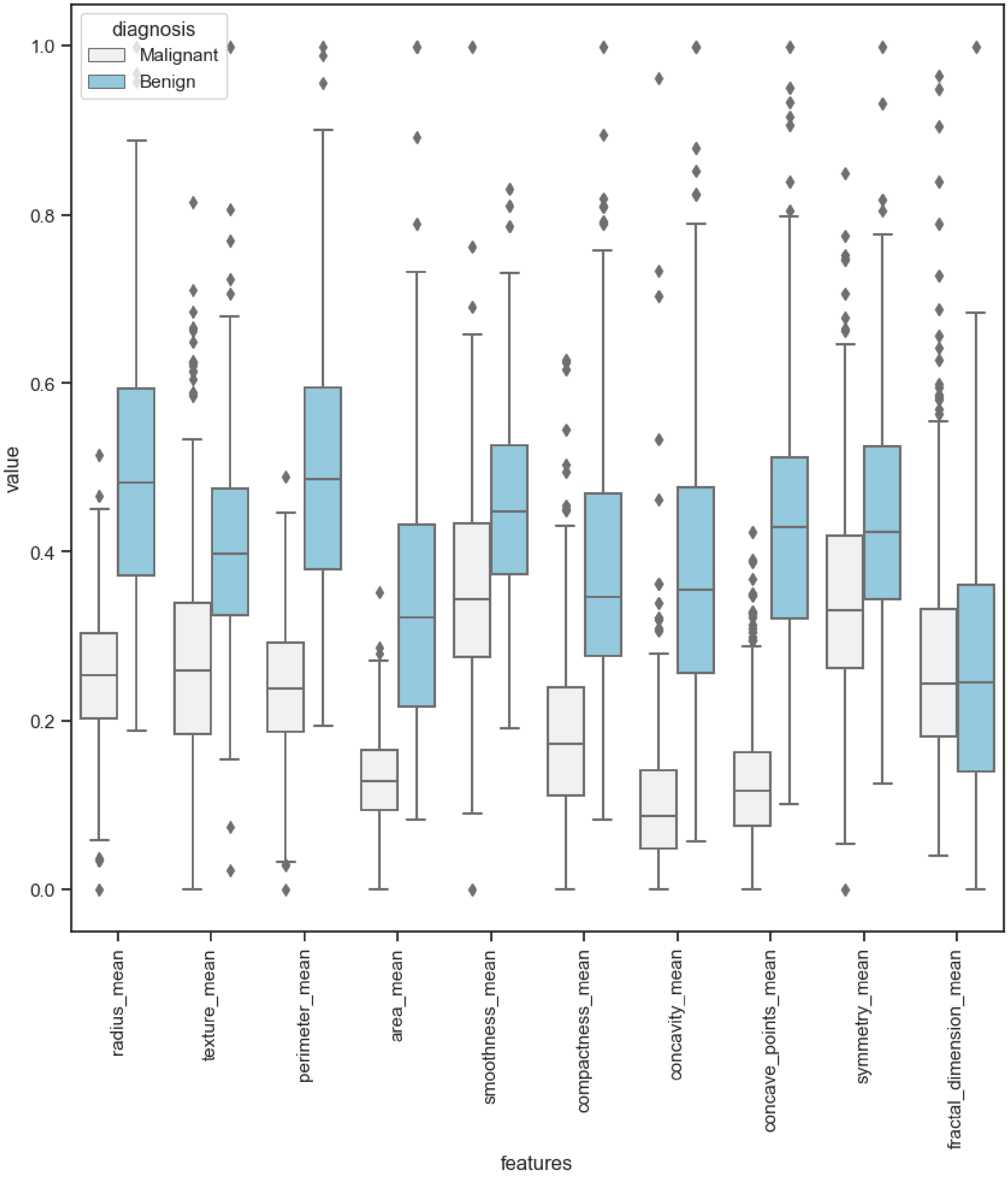
## Range of Values:

The minimum value is around 4.

The maximum value is roughly 28.



## Distribution plot of radial_mean



**Boxplot of certain features(radial_mean)**

## Observation:

1. The distribution is right-skewed.
2. The median is closer to the bottom of the box, indicating that half of the values fall below this point, while the other half are spread out over a wider range.
3. The whisker extending towards the higher values is longer than the whisker towards the lower values, further confirming the skewness.

## Histogram and boxplot to visualize the distribution of the data and detection of outliers

## Outliers affect analysis in the following ways:

- Outliers can significantly impact measures like mean, standard deviation, and range, making them less representative of the typical data.
- It can obscure patterns and relationships between variables in scatter plots or correlation analyses.
- Many statistical tests assume normality and homogeneity of variance, which can be violated by outliers, leading to unreliable results.

## #Getting the summary statistics of the area_mean feature using describe

**Input:**

```
df["area_mean"].describe()
```

Output:

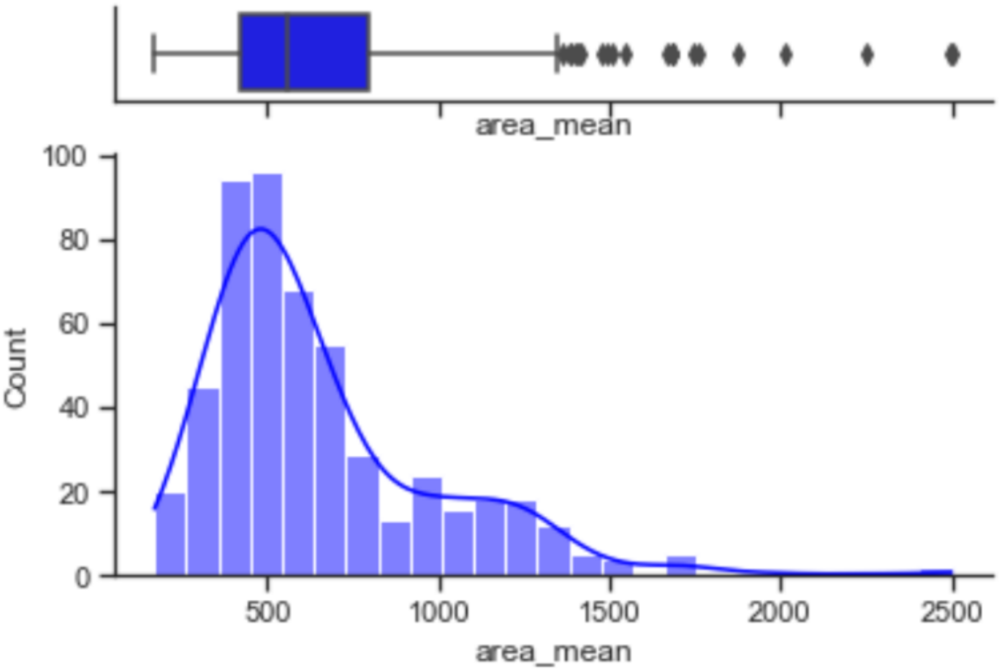
```
count    528.000000
mean     659.519697
std      351.435482
min      170.400000
25%      420.875000
50%      555.900000
75%      798.050000
max     2501.000000
```

Here the mean is **around 660** when the data is with outliers.

**Input:**

```
sns.set(style="ticks")
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True,
                                    gridspec_kw={"height_ratios": (.20, .90)})
sns.boxplot(df["area_mean"], ax =  ax_box , color = "blue")
sns.histplot(df["area_mean"],ax = ax_hist,  kde = True, color = "blue")
ax_box.set(yticks=[])
sns.despine(ax=ax_hist)
sns.despine(ax=ax_box)
```

**Output:**



## Observation:

1. The potential outlier for area_mean lies towards the end forming a tail in graph above 1500 (from value1500 to 2500)

**#Checking to see the effect of removal of extreme outliers**

Input:

```
df_outlierRemove = df.loc[df['area_mean'] < 1500]
df_outlierRemove["area_mean"].describe()
```

**In the above command :**

df_outlierRemove =: This part assigns the result of the operation on the right side to a new DataFrame called df_outlierRemove.

df.loc[]: This is a label-based data selection method in pandas, used to filter rows based on specific criteria.

df['area_mean'] < 1500: This is the condition being applied to filter the rows. It selects only the rows where the value in the area_mean column is less than 1500.

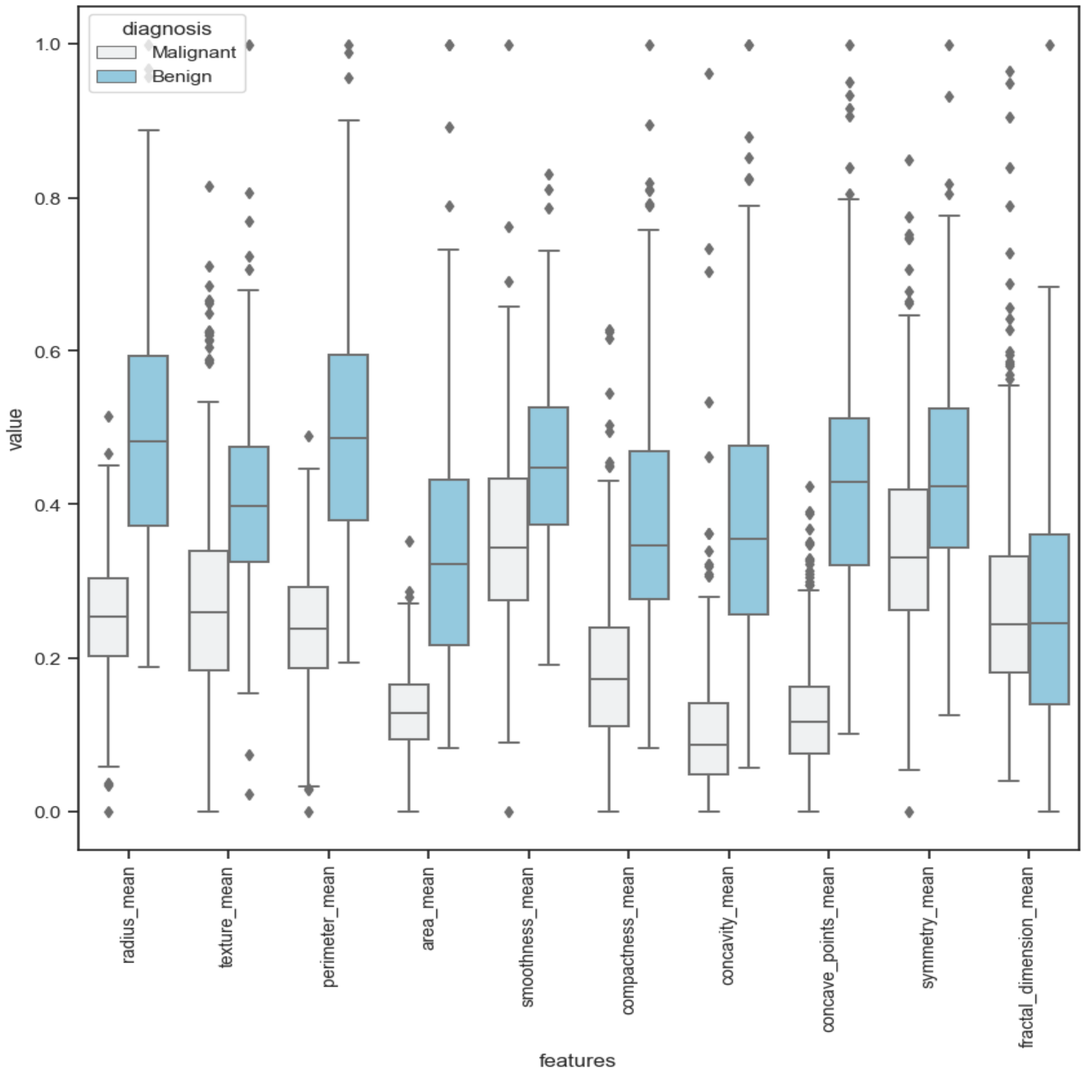
**Output:**

```
count     515.000000
mean      628.742524
std       292.288397
min       170.400000
25%       420.050000
50%       546.300000
75%       759.950000
max      1491.000000
Name: area_mean, dtype: float64
```

Here the mean is to **around 630** after removing outliers and filling then with null value

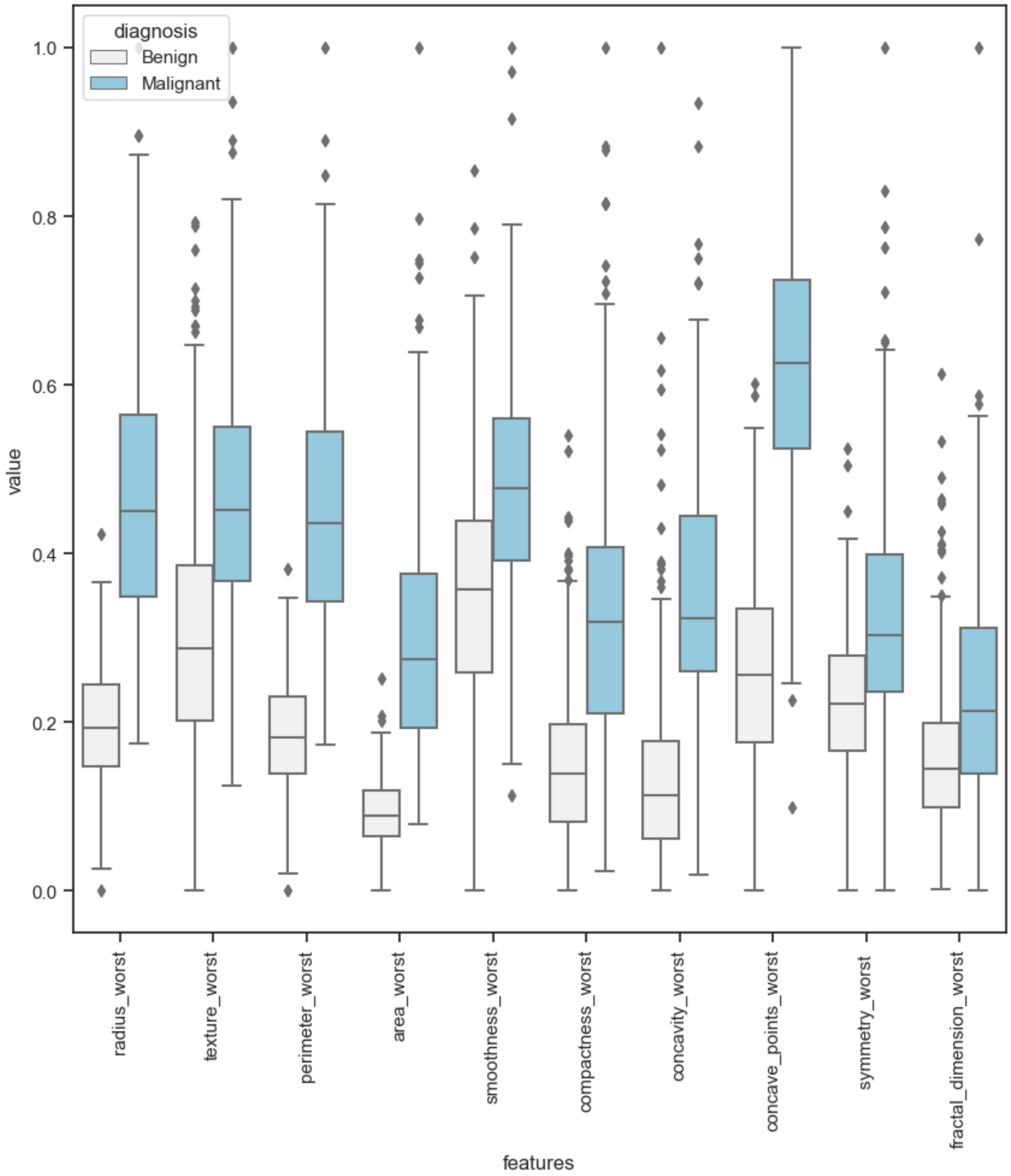
**The difference in mean before and after of removal of outliers is that a few very high values in "area_mean" can pull the mean upwards, even if most values are lower.**

## Factors most relevant predicting breast cancer diagnosis



**Observation:**

- radial_mean,texture_mean,perimeter_mean,area_mean,compactness_mean,concavity_mean,concave_points_mean.These features show significant differences between benign and malignant cases, with higher averages in malignant cases.



**Observation:**

- Radius_worst,perimeter_worst,area_worst,compactness_worst,concavity_worst,,concavity_points_worst. These features show significant differences between benign and malignant cases, with higher averages in malignant cases.

These features seem to potentially differentiate between benign and malignant breast cancer cases.

## Limitations in Data that might affect Analysis:

- Checking for the count of unique values under "diagnosis"

  **Input**:

  ```
  df["diagnosis"].value_counts()
  ```

  **Output:**

  ```
  diagnosis
  0    357
  1    212
  Name: count, dtype: int64
  ```

  Here 0 represents benign cases and 1 represents malignant cases. So the dataset has the imbalanced distribution of classes

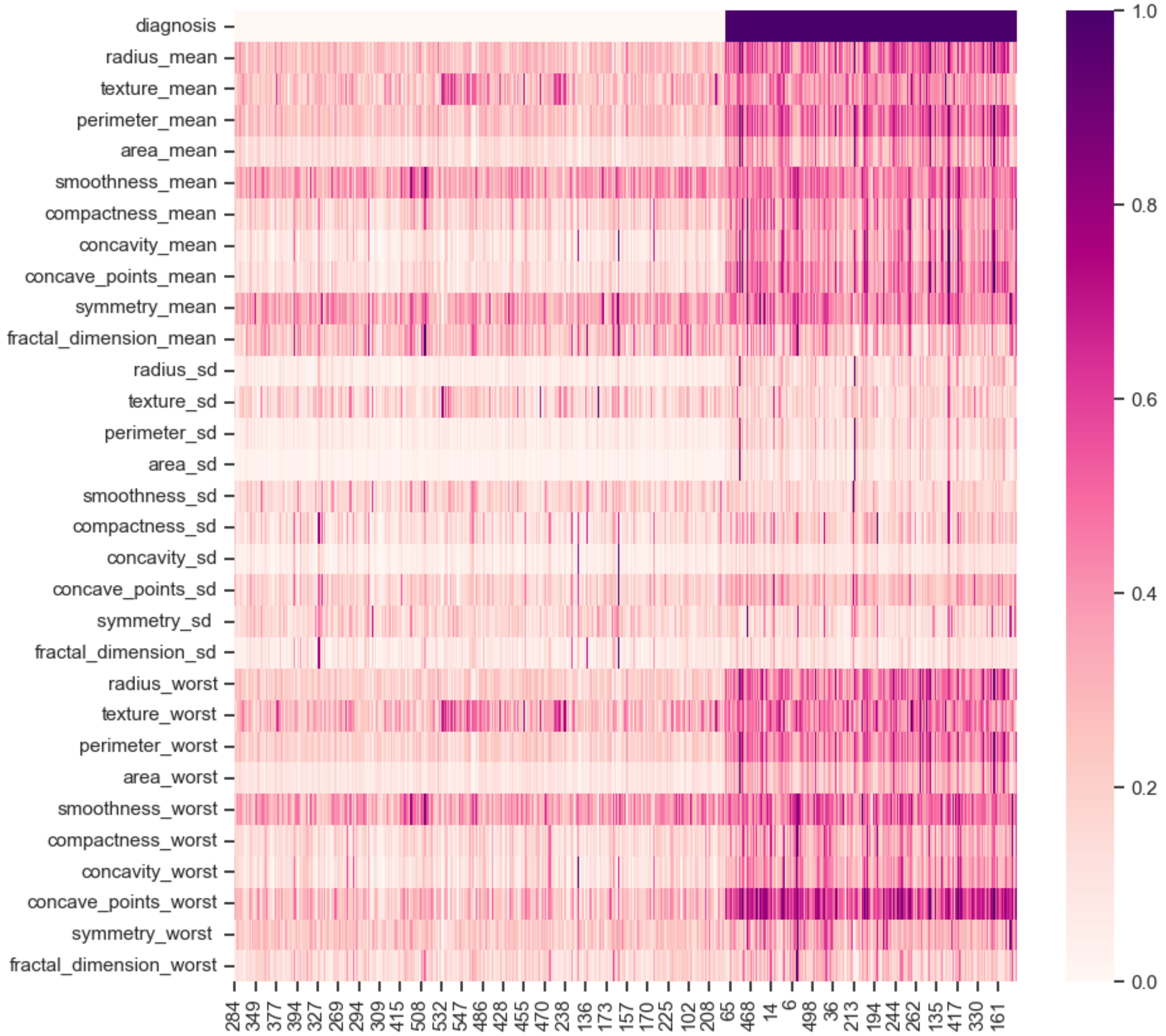
  Machine learning models trained on this data might be biased towards the majority class (benign), potentially leading to underestimating the malignant cases.

- Heatmap of the normalised data with features in rows and observations in columns. Reordering the columns such that the samples from the same class are grouped together.

  **#Reordering the columns such that the samples from the same class are grouped together**

  **Input**:

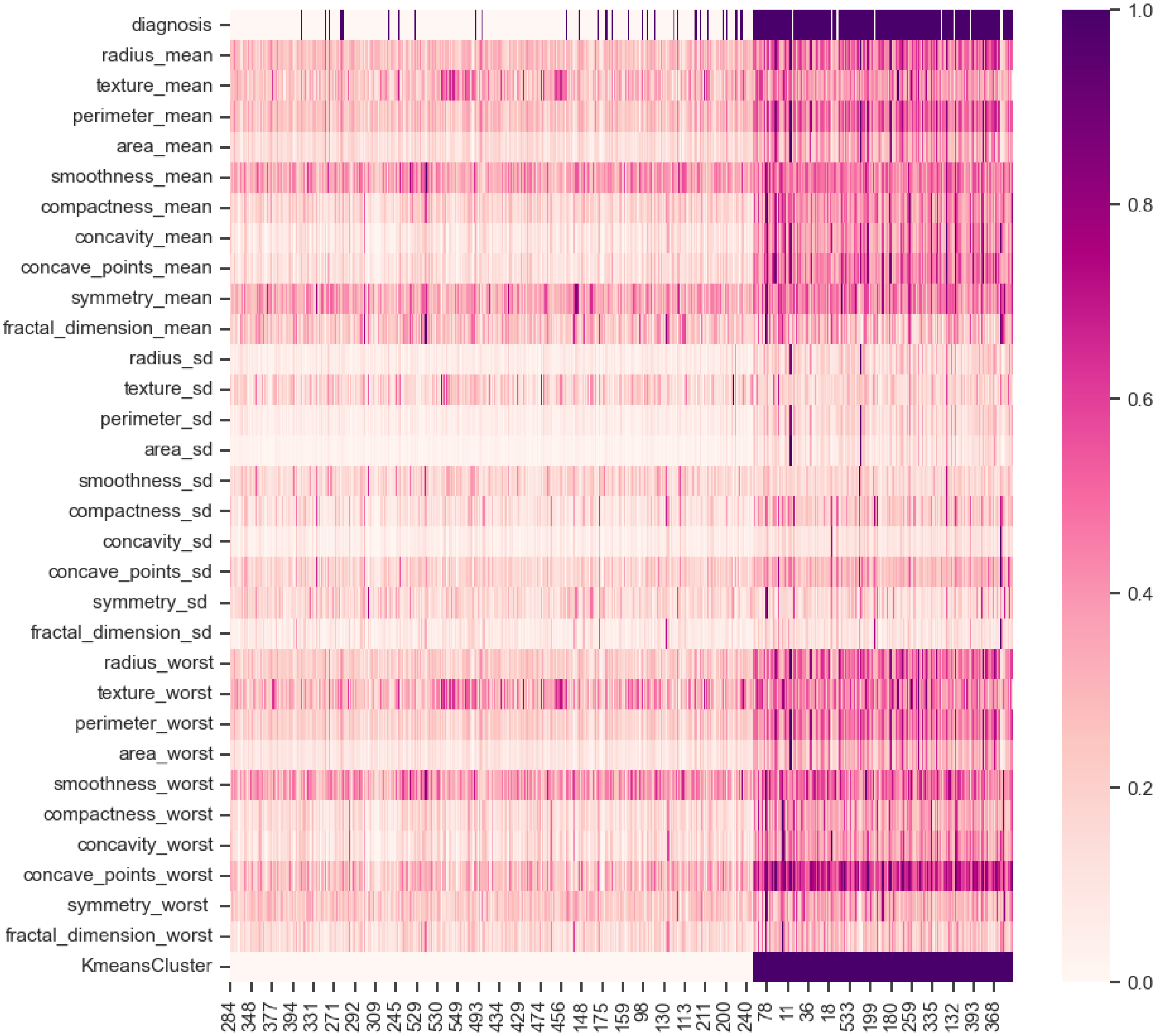
  ```
  fig, ax = plt.subplots(figsize=(10,10))
  NormSortDf = norm_df.sort_values(by = 'diagnosis')
  sns.heatmap(NormSortDf.T, cmap = "RdPu")
  ```



**#k-means clustering on the rows and rearranging them so that features from the same group are together**

**Input:**

```
from sklearn.cluster import KMeans
kmeans = KMeans(2)
kmeans.fit(norm_df.iloc[:,1:]) #Includes only features; Excludes "diagnosis"
identified_clusters = kmeans.fit_predict(norm_df.iloc[:,1:])
len(identified_clusters)
norm_df["KmeansCluster"] = identified_clusters
norm_dfKMSort = norm_df.sort_values("KmeansCluster")
fig, ax = plt.subplots(figsize=(10,10))
sns.heatmap(norm_dfKMSort.T, cmap= "RdPu")
```

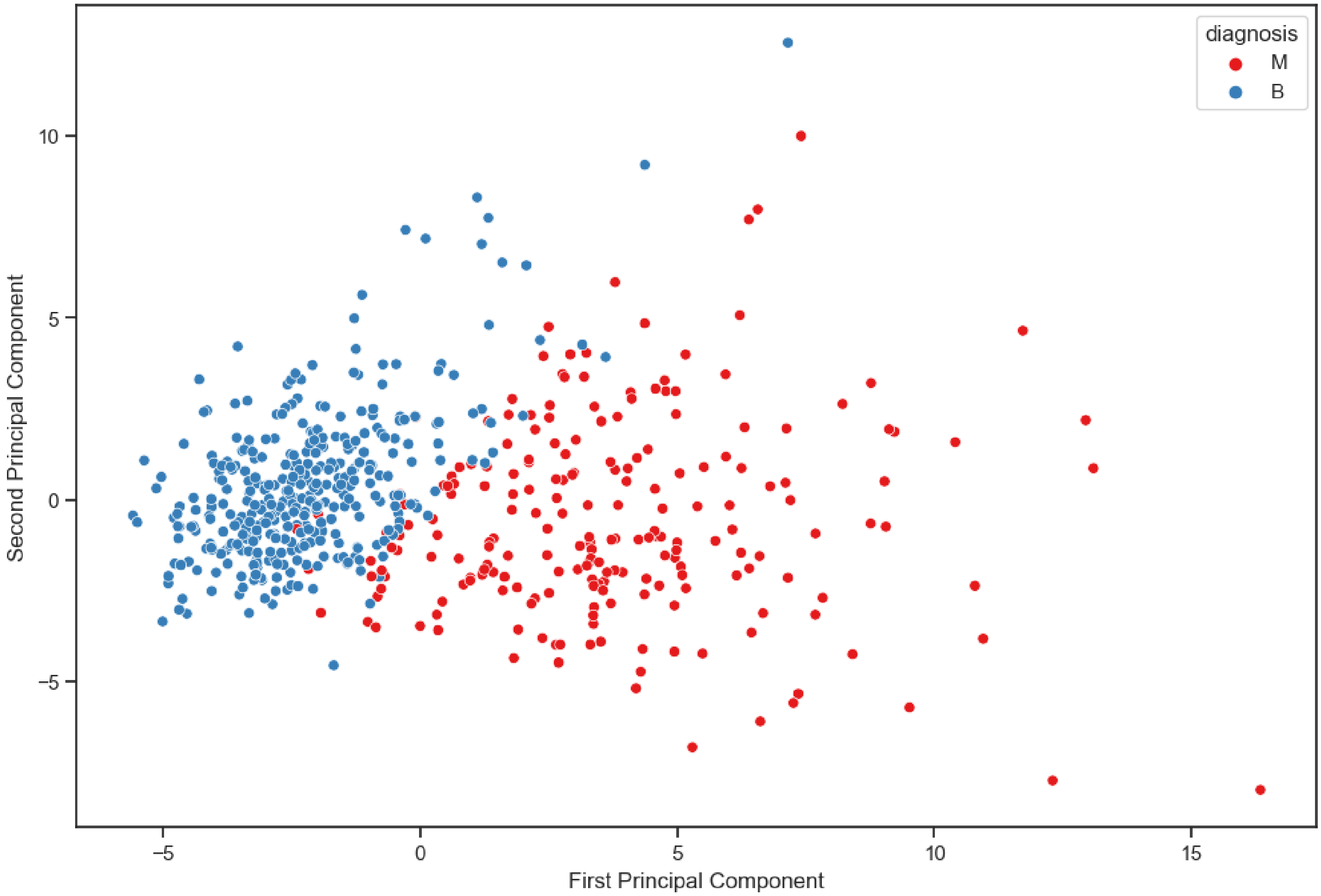


Rearranging columns based on the diagnosis feature and plotting a heat map of the reordered data shows that the "_mean" and the "_worst" features show clear difference between the two classes whereas the "_sd" features don't show much of a difference in the heatmap.

The heatmap created from the sorted diagnosis class and Kmeans cluster classified data shows a uniform heatmap indicating relationship between the features and the diagnosis class of the data.

- SDs offer insight into the spread or variability of data points around the mean. Outliers can significantly inflate SDs, distorting the representation of typical data variations.from the data given we can observe that some features(ex:area_mean) have potential outliers which affect the mean and variance values which need to be given attention during data cleaning and preprocessing.

- Principal Component Analysis Plot



According to the plot, the first principal component separates the data into two clusters, the left cluster is for malignant samples whereas as the right cluster is for benign samples. The benign sample data seems to be clustered together well except for a few outliers. The malignant cluster on the other hand seems more spread out and distant.

The data of benign and Malignant seem to overlap and are not distinct and no definite cluster group of two components(benign and malignant) and the malignant data is so scattered that this suggests great variability in features for malignant.

## Referance:

1.Box Plot Explained: Interpretation, Examples, & Comparison
BySaul Mcleod, PhD,Updated onJuly 31, 2023
2.Exploratory Data Analysis — Breast Cancer Wisconsin (Diagnostic) Dataset
Shashmi Karanam,Oct 1, 2022
3.https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data.
4.https://github.com/shashmi-jaiswal/EDA-for-Wisconsin-Breast-Cancer-Dagnostic-Dataset