colab.research.google.com/drive/1JXDKQi_10-IB8J3ArI_Cp9SPDypOswiH

breast cancer .ipynb

File Edit View Insert Runtime Tools Help Last edited on March 13

Comment Share

+ Code + Text

Connect Colab AI

About dataset

This dataset primarily focuses on the diagnosis of breast cancer, specifically distinguishing between malignant (cancerous) and benign (non-cancerous) tumors based on various attributes derived from digitized images of fine needle aspirates (FNA) of breast masses.The dataset typically contains records for several hundred patients...

+ Code + Text

```
[ ] import numpy as np
    import pandas as pd
    df=pd.read_csv('/content/data.csv')
    df
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	te
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710	...	
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017	...	
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790	...	
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520	...	
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.19800	0.10430	...	

ENG 00:30

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH

breast cancer .ipynb

File Edit View Insert Runtime Tools Help Last edited on March 13

Comment Share

+ Code + Text Connect Colab AI

```
df.head()
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	texti
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	...	
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	...	
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	...	
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	...	
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	...	

5 rows × 33 columns

```
[ ] df.tail()
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	texti
564	926424	M	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	...	

ENG 00:30

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH

breast cancer .ipynb

File Edit View Insert Runtime Tools Help Last edited on March 13

Comment Share

+ Code + Text Connect Colab AI

```
df.tail()
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	texti
564	926424	M	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	...	
565	926682	M	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791	...	
566	926954	M	16.60	28.08	108.30	858.1	0.08455	0.10230	0.09251	0.05302	...	
567	927241	M	20.60	29.33	140.10	1265.0	0.11780	0.27700	0.35140	0.15200	...	
568	92751	B	7.76	24.54	47.92	181.0	0.05263	0.04362	0.00000	0.00000	...	

5 rows × 33 columns

```
[ ] df.dtypes
```

```
id                       int64
diagnosis               object
radius_mean            float64
texture_mean           float64
perimeter_mean         float64
area_mean              float64
```

ENG 00:30

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH

breast cancer .ipynb

File Edit View Insert Runtime Tools Help Last edited on March 13

Comment Share

+ Code + Text

Connect Colab AI

```
[ ] df.isna().sum()
```

```
id                        0
diagnosis                 0
radius_mean               0
texture_mean              0
perimeter_mean            0
area_mean                 0
smoothness_mean           0
compactness_mean          0
concavity_mean            0
concave points_mean       0
symmetry_mean             0
fractal_dimension_mean    0
radius_se                 0
texture_se                0
perimeter_se              0
area_se                   0
smoothness_se             0
compactness_se            0
concavity_se              0
concave points_se         0
symmetry_se               0
fractal_dimension_se      0
radius_worst              0
texture_worst             0
```

ENG 00:30

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH

breast cancer .ipynb

File Edit View Insert Runtime Tools Help Last edited on March 13

Comment Share

+ Code + Text

Connect Colab AI

```
df.dtypes
```

```
id                         int64
diagnosis                 object
radius_mean              float64
texture_mean             float64
perimeter_mean           float64
area_mean                float64
smoothness_mean          float64
compactness_mean         float64
concavity_mean           float64
concave points_mean      float64
symmetry_mean            float64
fractal_dimension_mean   float64
radius_se                float64
texture_se               float64
perimeter_se             float64
area_se                  float64
smoothness_se            float64
compactness_se           float64
concavity_se             float64
concave points_se        float64
symmetry_se              float64
fractal_dimension_se     float64
radius_worst             float64
texture_worst            float64
perimeter_worst          float64
```

ENG 00:30

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH

breast cancer .ipynb

File Edit View Insert Runtime Tools Help Last edited on March 13

Comment Share

+ Code + Text

Connect Colab AI

```
[ ]  9   concave points_mean      569 non-null    float64
     10  symmetry_mean            569 non-null    float64
     11  fractal_dimension_mean   569 non-null    float64
     12  radius_se                569 non-null    float64
     13  texture_se               569 non-null    float64
     14  perimeter_se             569 non-null    float64
     15  area_se                  569 non-null    float64
     16  smoothness_se            569 non-null    float64
     17  compactness_se           569 non-null    float64
     18  concavity_se             569 non-null    float64
     19  concave points_se        569 non-null    float64
     20  symmetry_se              569 non-null    float64
     21  fractal_dimension_se     569 non-null    float64
     22  radius_worst             569 non-null    float64
     23  texture_worst            569 non-null    float64
     24  perimeter_worst          569 non-null    float64
     25  area_worst               569 non-null    float64
     26  smoothness_worst         569 non-null    float64
     27  compactness_worst        569 non-null    float64
     28  concavity_worst          569 non-null    float64
     29  concave points_worst     569 non-null    float64
     30  symmetry_worst           569 non-null    float64
     31  fractal_dimension_worst  569 non-null    float64
     32  Unnamed: 32              0 non-null      float64
    dtypes: float64(31), int64(1), object(1)
    memory usage: 146.8+ KB
```

ENG 00:30

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH

breast cancer .ipynb

File Edit View Insert Runtime Tools Help Last edited on March 13

Comment Share

+ Code + Text

Connect Colab AI

```
[ ] df.columns
```

```
Index(['id', 'diagnosis', 'radius_mean', 'texture_mean', 'perimeter_mean',
       'area_mean', 'smoothness_mean', 'compactness_mean', 'concavity_mean',
       'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean',
       'radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
       'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
       'fractal_dimension_se', 'radius_worst', 'texture_worst',
       'perimeter_worst', 'area_worst', 'smoothness_worst',
       'compactness_worst', 'concavity_worst', 'concave points_worst',
       'symmetry_worst', 'fractal_dimension_worst', 'Unnamed: 32'],
      dtype='object')
```

```
[ ] df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 33 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   id                      569 non-null    int64
 1   diagnosis               569 non-null    object
 2   radius_mean             569 non-null    float64
 3   texture_mean            569 non-null    float64
 4   perimeter_mean          569 non-null    float64
```

ENG 00:30

colab.research.google.com/drive/1JXDKQi_10-IB8J3ArI_Cp9SPDypOswiH

breast cancer .ipynb

File Edit View Insert Runtime Tools Help Last edited on March 13

Comment Share

+ Code + Text Connect Colab AI

```
df.drop(['id','Unnamed: 32'],axis=1,inplace=True)
df
```

	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	..
0	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710	0.2419	..
1	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017	0.1812	..
2	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790	0.2069	..
3	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520	0.2597	..
4	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.19800	0.10430	0.1809	..
...	...	...	...	...	...	...	...	...	...	...	..
564	M	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	0.1726	..
565	M	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791	0.1752	..
566	M	16.60	28.08	108.30	858.1	0.08455	0.10230	0.09251	0.05302	0.1590	..
567	M	20.60	29.33	140.10	1265.0	0.11780	0.27700	0.35140	0.15200	0.2397	..
568	B	7.76	24.54	47.92	181.0	0.05263	0.04362	0.00000	0.00000	0.1587	..

569 rows × 31 columns

ENG 00:30

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH

breast cancer .ipynb

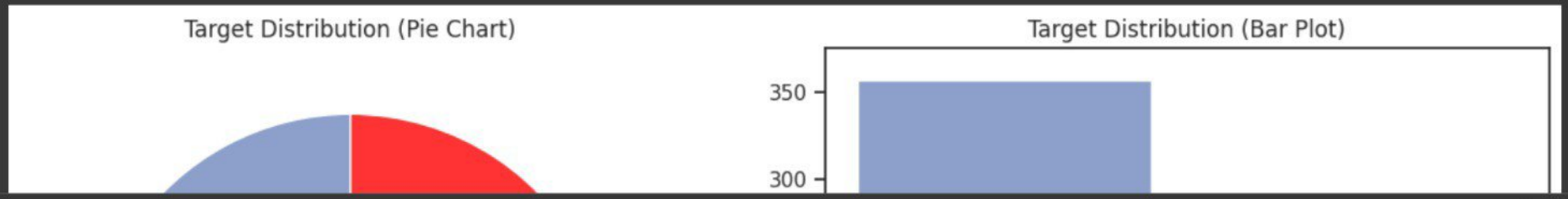
File Edit View Insert Runtime Tools Help Last edited on March 13

Comment Share

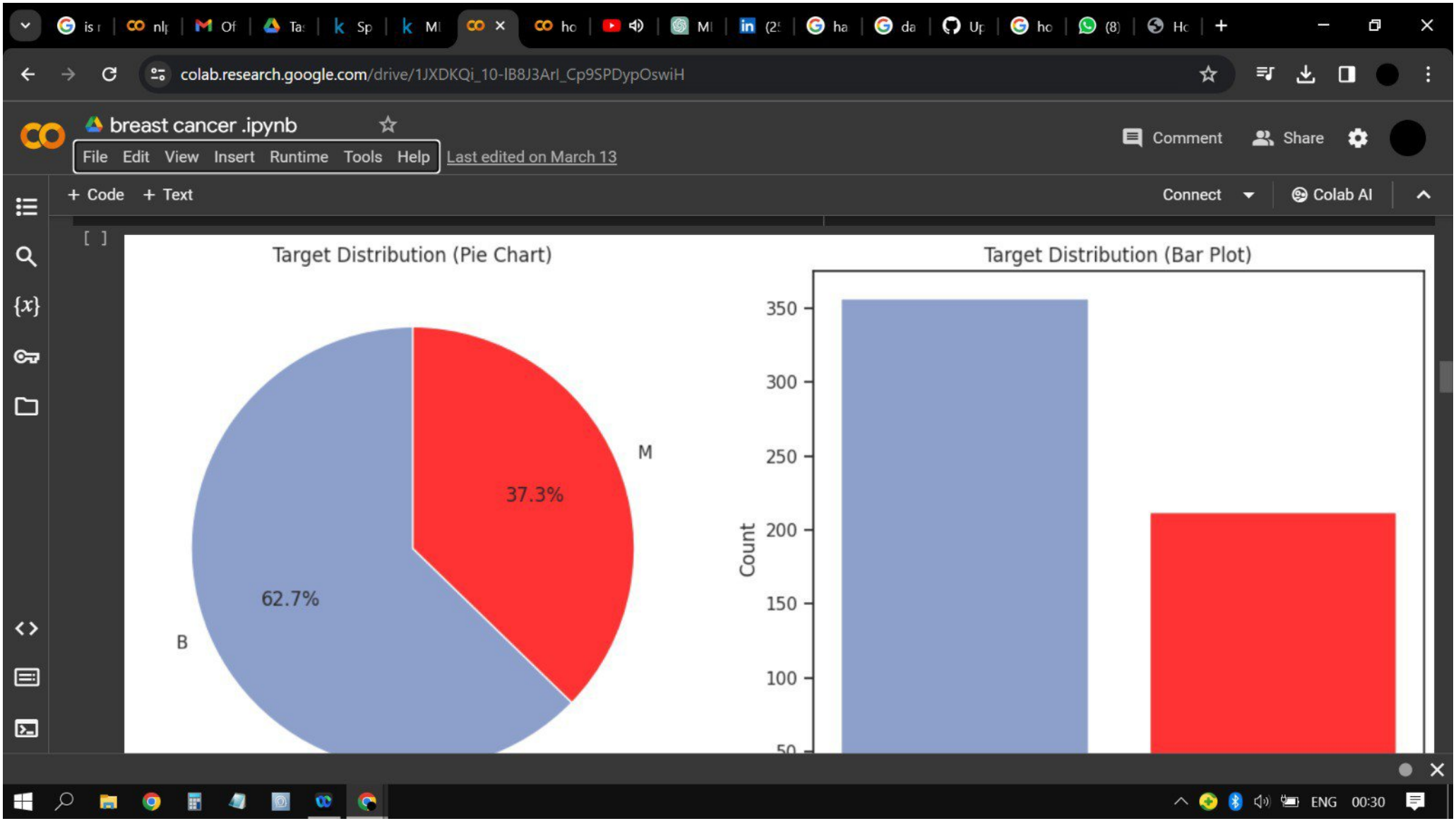
+ Code + Text

Connect Colab AI

```
target_count = df['diagnosis'].value_counts()
pie_colors = ['#8da0cb', '#FF3333']
bar_colors = ['#8da0cb', '#FF3333']
fig, axs = plt.subplots(1, 2, figsize=(12, 6))
# Plot the pie chart on the first subplot with custom colors
axs[0].pie(target_count, labels=target_count.index, autopct='%1.1f%%', startangle=90, colors=pie_colors)
axs[0].set_title('Target Distribution (Pie Chart)')
# Plot the bar plot on the second subplot with custom colors
axs[1].bar(target_count.index, target_count.values, color=bar_colors)
axs[1].set_xticks(target_count.index)
axs[1].set_xticklabels(['Benign', 'Malignant'])
axs[1].set_xlabel('Target')
axs[1].set_ylabel('Count')
axs[1].set_title('Target Distribution (Bar Plot)')
plt.tight_layout()
plt.show()
```



ENG 00:30


colab.research.google.com/drive/1JXDKQi_10-IB8J3ArI_Cp9SPDypOswiH
breast cancer .ipynb
File Edit View Insert Runtime Tools Help
Last edited on March 13
Comment
Share
+ Code + Text
Connect
Colab AI
Target Distribution (Pie Chart)
M
37.3%
62.7%
B
Target Distribution (Bar Plot)
Count
350
300
250
200
150
100
50
ENG 00:30

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

breast cancer .ipynb

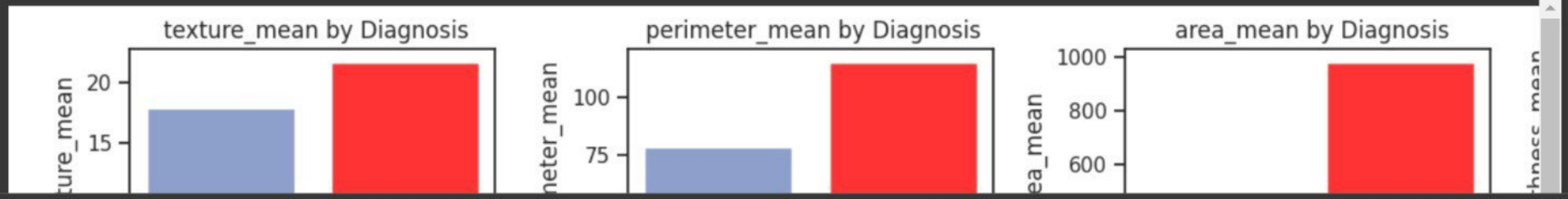
File Edit View Insert Runtime Tools Help Last edited on March 13

Comment Share

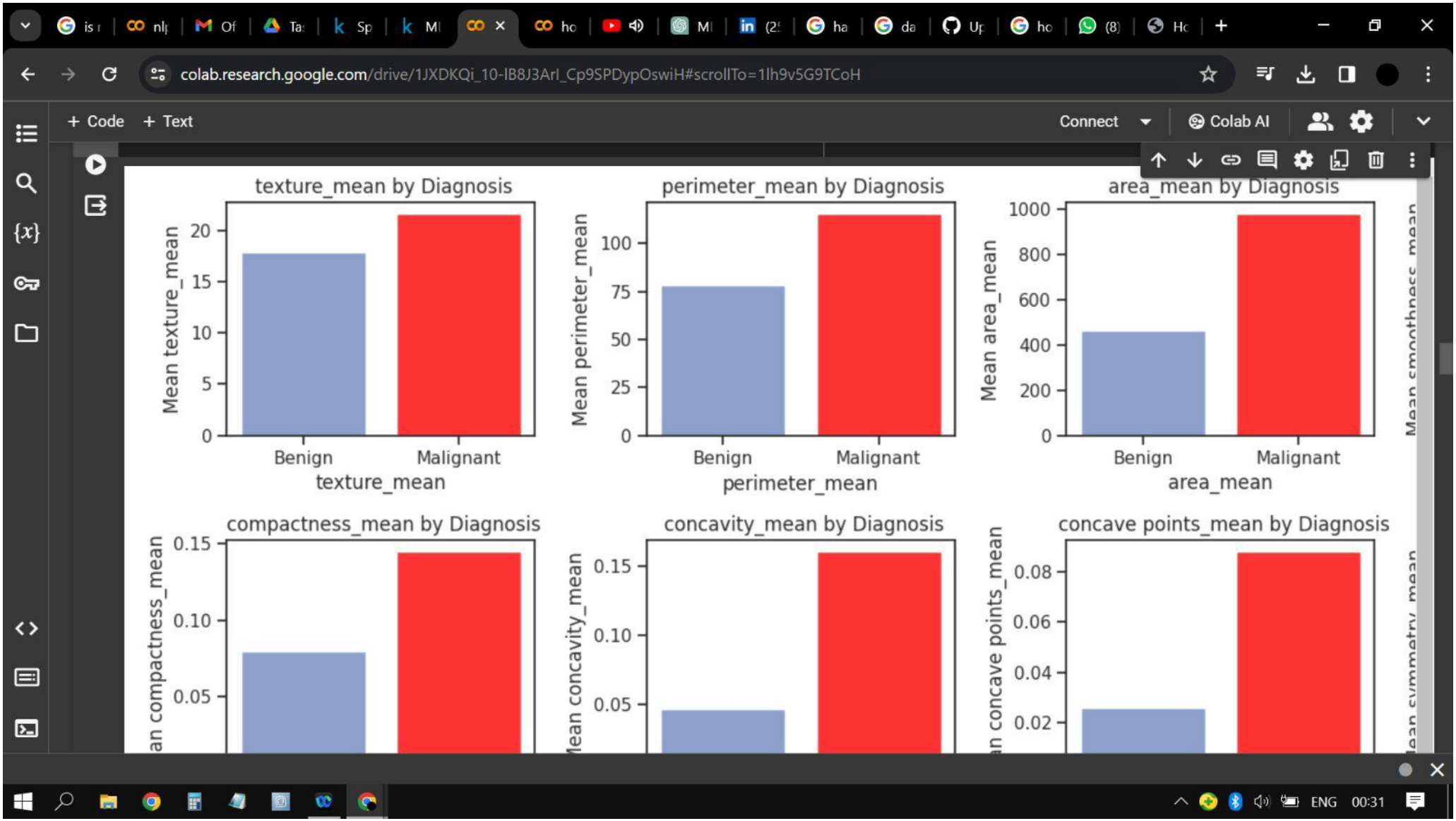
+ Code + Text

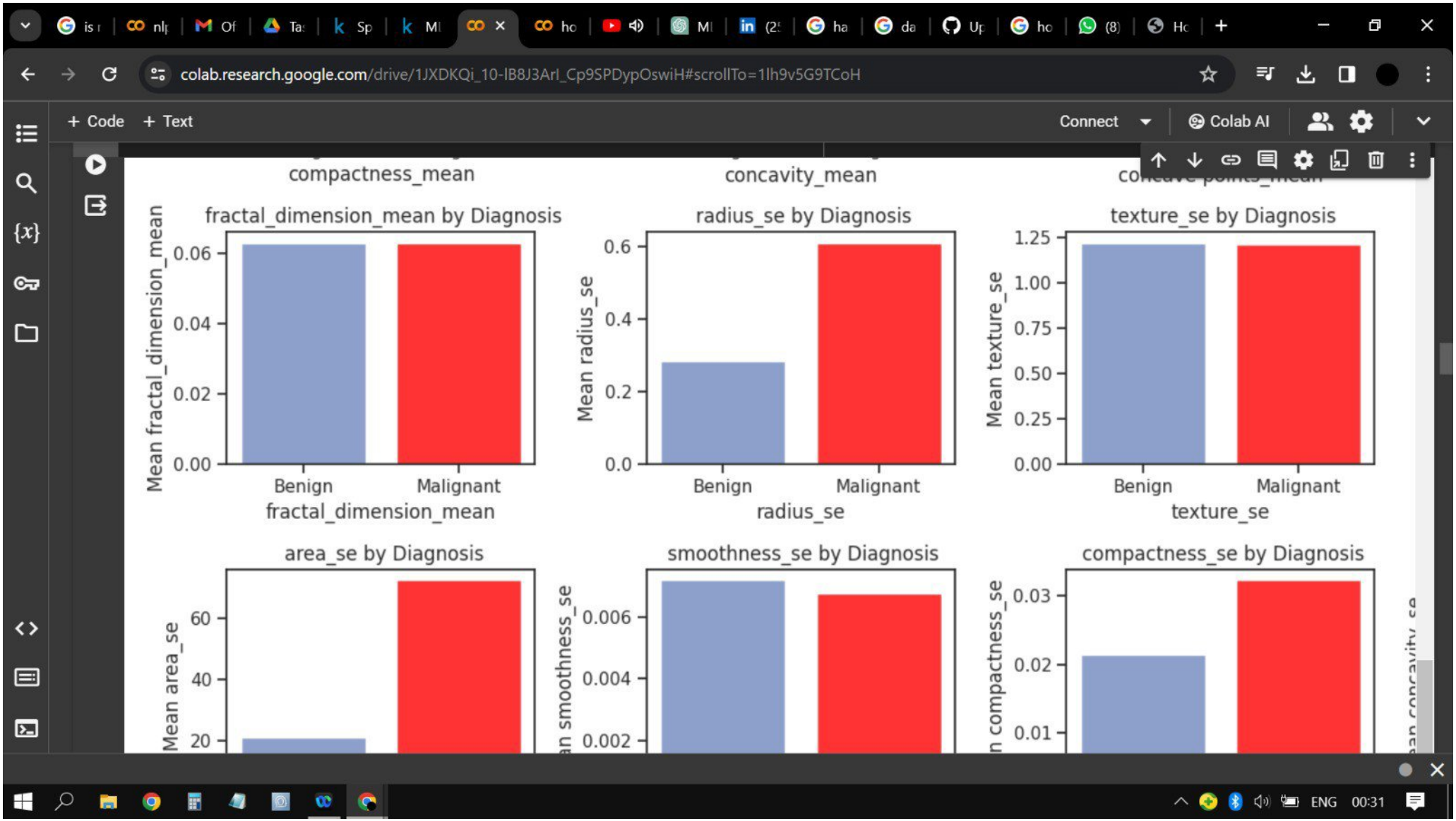
Connect Colab AI

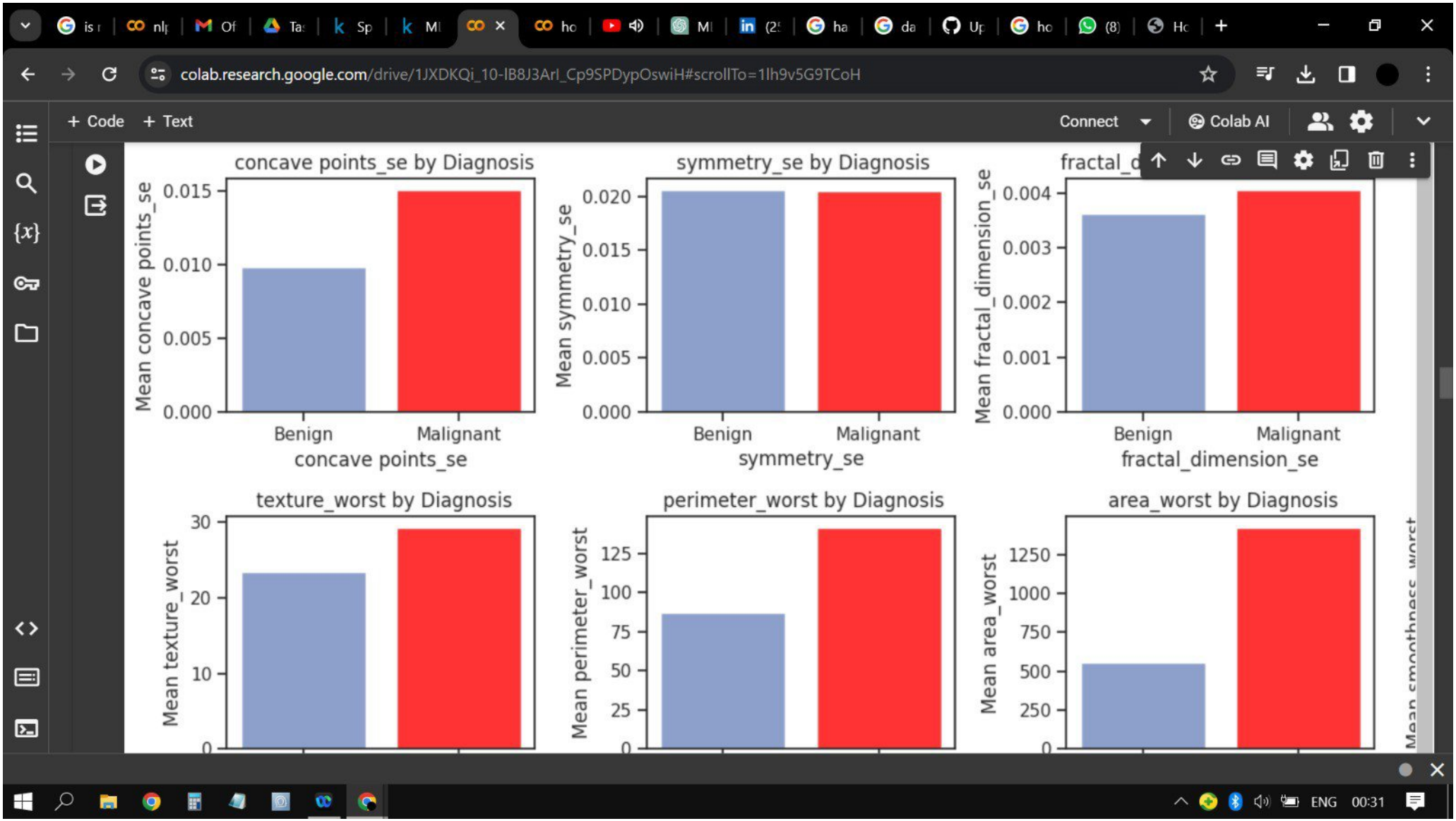
```
num_plots = len(df.columns[2:32])
num_cols = 4
num_rows = (num_plots // num_cols) + (num_plots % num_cols)

plt.figure(figsize=(15, 3*num_rows))
for i, col in enumerate(df.columns[2:32], 1):
    plt.subplot(num_rows, num_cols, i)
    value_mean = df.groupby('diagnosis')[col].mean()
    plt.bar(value_mean.index, value_mean.values, color=['#8da0cb', '#FF3333'])
    plt.xlabel(col)
    plt.ylabel(f'Mean {col}')
    plt.title(f' {col} by Diagnosis')
    plt.xticks(value_mean.index, ['Benign', 'Malignant'])
plt.tight_layout()
plt.show()
```

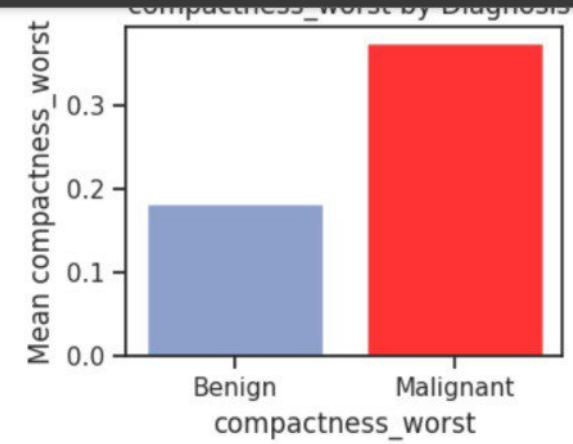


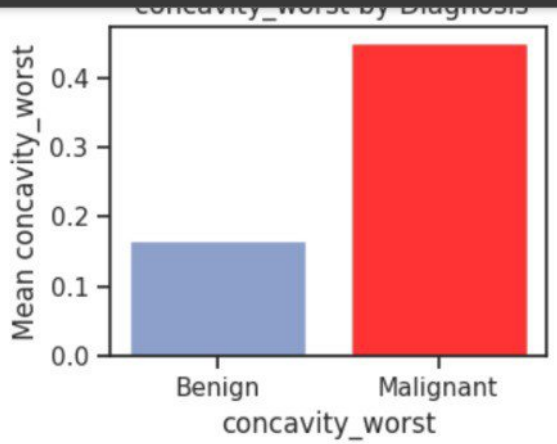
ENG 00:31

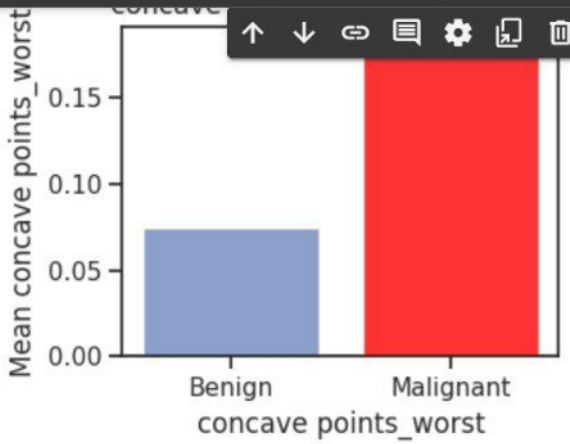

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH
+ Code
+ Text
Connect
Colab AI
texture_mean by Diagnosis
Mean texture_mean
Benign
Malignant
texture_mean
perimeter_mean by Diagnosis
Mean perimeter_mean
Benign
Malignant
perimeter_mean
area_mean by Diagnosis
Mean area_mean
Benign
Malignant
area_mean
compactness_mean by Diagnosis
concavity_mean by Diagnosis
concave points_mean by Diagnosis
ENG
00:31

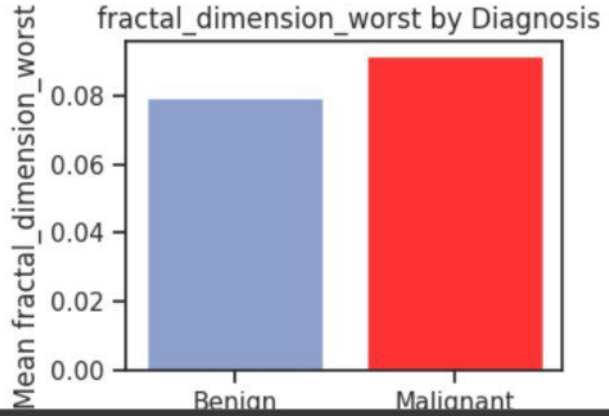

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH
+ Code
+ Text
Connect
Colab AI
compactness_mean
concavity_mean
fractal_dimension_mean by Diagnosis
Mean fractal_dimension_mean
0.06
0.04
0.02
0.00
Benign
Malignant
fractal_dimension_mean
radius_se by Diagnosis
Mean radius_se
0.6
0.4
0.2
0.0
Benign
Malignant
radius_se
texture_se by Diagnosis
Mean texture_se
1.25
1.00
0.75
0.50
0.25
0.00
Benign
Malignant
texture_se
area_se by Diagnosis
Mean area_se
60
40
20
smoothness_se by Diagnosis
0.006
0.004
0.002
compactness_se by Diagnosis
0.03
0.02
0.01
ENG
00:31


colab.research.google.com/drive/1JXDKQi_10-IB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH
+ Code
+ Text
Connect
Colab AI
concave points_se by Diagnosis
Mean concave points_se
0.015
0.010
0.005
0.000
Benign
Malignant
concave points_se
symmetry_se by Diagnosis
Mean symmetry_se
0.020
0.015
0.010
0.005
0.000
Benign
Malignant
symmetry_se
fractal_d
Mean fractal_dimension_se
0.004
0.003
0.002
0.001
0.000
Benign
Malignant
fractal_dimension_se
texture_worst by Diagnosis
Mean texture_worst
30
20
10
0
perimeter_worst by Diagnosis
Mean perimeter_worst
125
100
75
50
25
0
area_worst by Diagnosis
Mean area_worst
1250
1000
750
500
250
0
ENG
00:31


Mean compactness_worst
0.0
0.1
0.2
0.3
Benign
Malignant
compactness_worst


Mean concavity_worst
0.0
0.1
0.2
0.3
0.4
Benign
Malignant
concavity_worst


Mean concave points_worst
0.00
0.05
0.10
0.15
Benign
Malignant
concave points_worst

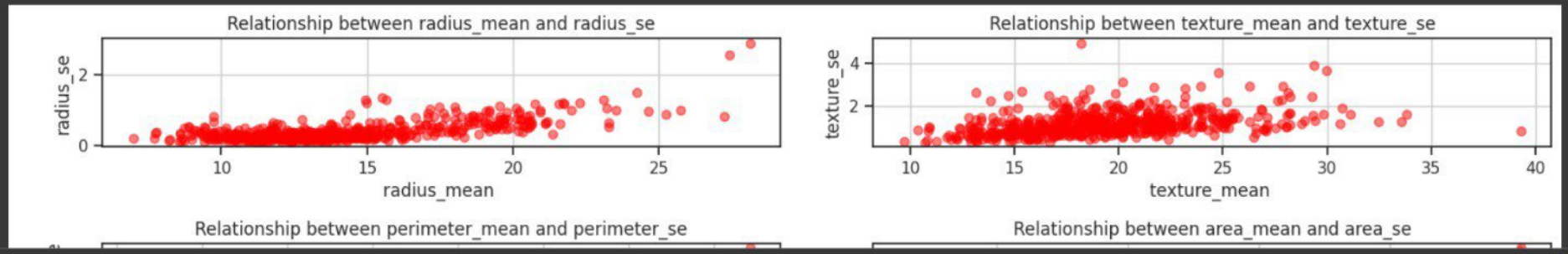

fractal_dimension_worst by Diagnosis
Mean fractal_dimension_worst
0.00
0.02
0.04
0.06
0.08
Benign
Malignant

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```python
num_rows = num_features // 2
num_cols = 2 if num_features % 2 == 0 else num_features // 2 + 1

fig, axs = plt.subplots(num_rows, num_cols, figsize=(15, 10))
# Flatten the axs array if it's more than 1D
if num_rows > 1:
    axs = axs.flatten()
# Iterate over each feature in df_se and df_mean
for i, (feature_se, feature_mean) in enumerate(zip(df_se.columns, df_mean.columns)):
    # Create a scatter plot on the current axis
    axs[i].scatter(df_mean[feature_mean], df_se[feature_se], color='red',alpha=0.5)
    axs[i].set_xlabel(f'{feature_mean} ')
    axs[i].set_ylabel(f'{feature_se}')
    axs[i].set_title(f'Relationship between {feature_mean} and {feature_se}')
    axs[i].grid(True)
# Adjust layout
plt.tight_layout()
plt.show()
```



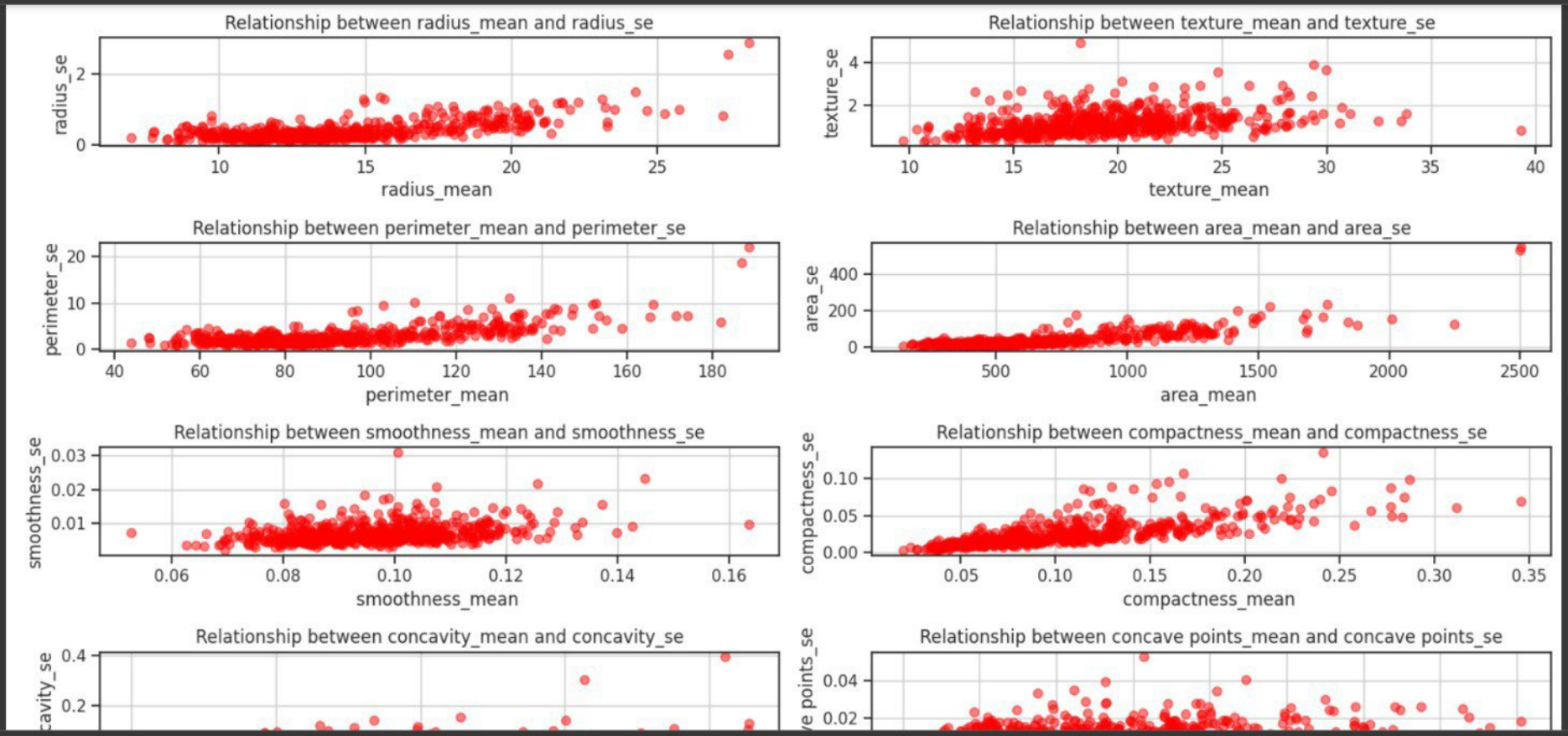
ENG 00:31

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

[ ]



ENG 00:31

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
df.groupby("diagnosis").max().style.background_gradient(cmap='coolwarm')
```

diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fracta
B	17.850000	33.810000	114.600000	992.100000	0.163400	0.223900	0.410800	0.085340	0.274300	
M	28.110000	39.280000	188.500000	2501.000000	0.144700	0.345400	0.426800	0.201200	0.304000	

```
[ ] # Relation between Mean Features and their standard eror
    df_se = df[['radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
           'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
           'fractal_dimension_se']]
    df_mean = df[['radius_mean', 'texture_mean', 'perimeter_mean',
           'area_mean', 'smoothness_mean', 'compactness_mean', 'concavity_mean',
           'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean']]
```

```
[ ] num_features = len(df_se.columns)
    num_rows = num_features // 2
    num_cols = 2 if num_features % 2 == 0 else num_features // 2 + 1

    fig, axs = plt.subplots(num_rows, num_cols, figsize=(15, 10))
    # Flatten the axs array if it's more than 1D
```

ENG 00:31

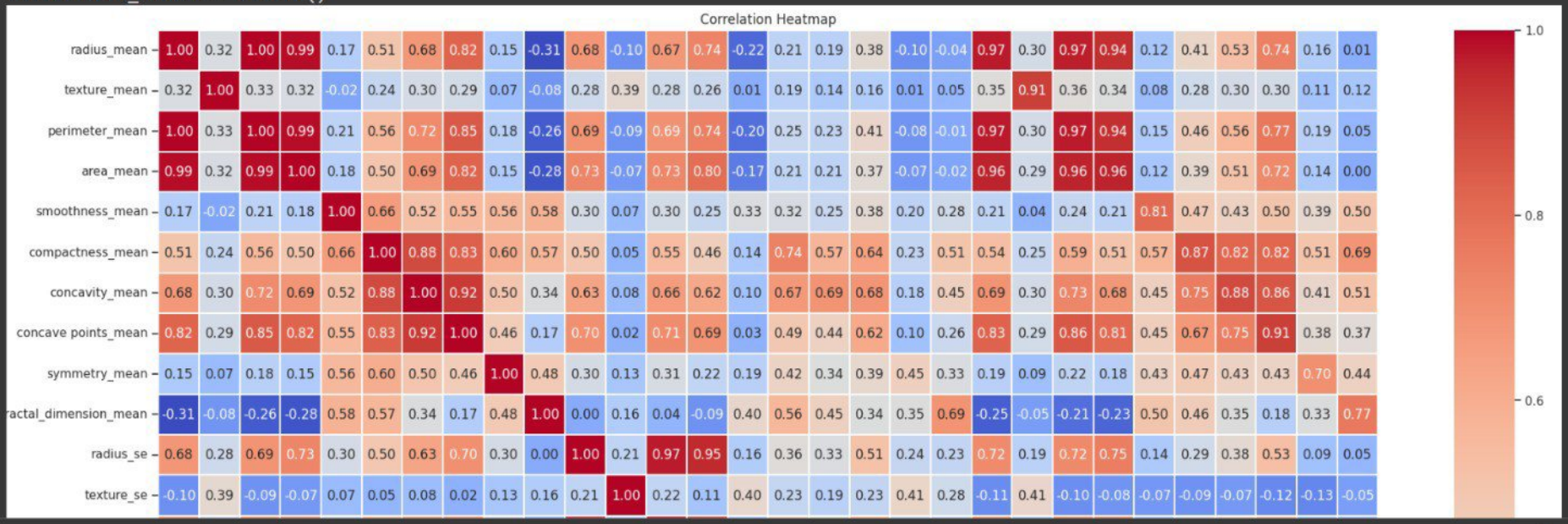
colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
correlation_matrix = df.corr()
plt.figure(figsize=(25, 20))
sns.heatmap(correlation_matrix, annot=True, linewidths=0.3, linecolor="white", fmt=".2f", cmap='coolwarm')
plt.title("Correlation Heatmap")
plt.show()
```

```
ipython-input-88-11ba5ba9020a>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will defaul
  correlation_matrix = df.corr()
```



ENG 00:31

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

area_se	0.74	0.26	0.74	0.80	0.25	0.46	0.62	0.69	0.22	-0.09	0.95	0.11	0.94	1.00	0.08	0.28	0.27	0.42	0.13	0.13	0.76	0.20	0.76	0.81	0.13	0.28	0.39	0.54	0.07	0.02
smoothness_se	-0.22	0.01	-0.20	-0.17	0.33	0.14	0.10	0.03	0.19	0.40	0.16	0.40	0.15	0.08	1.00	0.34	0.27	0.33	0.41	0.43	-0.23	-0.07	-0.22	-0.18	0.31	-0.06	-0.06	-0.10	-0.11	0.10
compactness_se	0.21	0.19	0.25	0.21	0.32	0.74	0.67	0.49	0.42	0.56	0.36	0.23	0.42	0.28	0.34	1.00	0.80	0.74	0.39	0.80	0.20	0.14	0.26	0.20	0.23	0.68	0.64	0.48	0.28	0.59
concavity_se	0.19	0.14	0.23	0.21	0.25	0.57	0.69	0.44	0.34	0.45	0.33	0.19	0.36	0.27	0.27	0.80	1.00	0.77	0.31	0.73	0.19	0.10	0.23	0.19	0.17	0.48	0.66	0.44	0.20	0.44
concave points_se	0.38	0.16	0.41	0.37	0.38	0.64	0.68	0.62	0.39	0.34	0.51	0.23	0.56	0.42	0.33	0.74	0.77	1.00	0.31	0.61	0.36	0.09	0.39	0.34	0.22	0.45	0.55	0.60	0.14	0.31
symmetry_se	-0.10	0.01	-0.08	-0.07	0.20	0.23	0.18	0.10	0.45	0.35	0.24	0.41	0.27	0.13	0.41	0.39	0.31	0.31	1.00	0.37	-0.13	-0.08	-0.10	-0.11	-0.01	0.06	0.04	-0.03	0.39	0.08
fractal_dimension_se	-0.04	0.05	-0.01	-0.02	0.28	0.51	0.45	0.26	0.33	0.69	0.23	0.28	0.24	0.13	0.43	0.80	0.73	0.61	0.37	1.00	-0.04	-0.00	-0.00	-0.02	0.17	0.39	0.38	0.22	0.11	0.59
radius_worst	0.97	0.35	0.97	0.96	0.21	0.54	0.69	0.83	0.19	-0.25	0.72	-0.11	0.70	0.76	-0.23	0.20	0.19	0.36	-0.13	-0.04	1.00	0.36	0.99	0.98	0.22	0.48	0.57	0.79	0.24	0.09
texture_worst	0.30	0.91	0.30	0.29	0.04	0.25	0.30	0.29	0.09	-0.05	0.19	0.41	0.20	0.20	-0.07	0.14	0.10	0.09	-0.08	-0.00	0.36	1.00	0.37	0.35	0.23	0.36	0.37	0.36	0.23	0.22
perimeter_worst	0.97	0.36	0.97	0.96	0.24	0.59	0.73	0.86	0.22	-0.21	0.72	-0.10	0.72	0.76	-0.22	0.26	0.23	0.39	-0.10	-0.00	0.99	0.37	1.00	0.98	0.24	0.53	0.62	0.82	0.27	0.14
area_worst	0.94	0.34	0.94	0.96	0.21	0.51	0.68	0.81	0.18	-0.23	0.75	-0.08	0.73	0.81	-0.18	0.20	0.19	0.34	-0.11	-0.02	0.98	0.35	0.98	1.00	0.21	0.44	0.54	0.75	0.21	0.08
smoothness_worst	0.12	0.08	0.15	0.12	0.81	0.57	0.45	0.45	0.43	0.50	0.14	-0.07	0.13	0.13	0.31	0.23	0.17	0.22	-0.01	0.17	0.22	0.23	0.24	0.21	1.00	0.57	0.52	0.55	0.49	0.62
compactness_worst	0.41	0.28	0.46	0.39	0.47	0.87	0.75	0.67	0.47	0.46	0.29	-0.09	0.34	0.28	-0.06	0.68	0.48	0.45	0.06	0.39	0.48	0.36	0.53	0.44	0.57	1.00	0.89	0.80	0.61	0.81
concavity_worst	0.53	0.30	0.56	0.51	0.43	0.82	0.88	0.75	0.43	0.35	0.38	-0.07	0.42	0.39	-0.06	0.64	0.66	0.55	0.04	0.38	0.57	0.37	0.62	0.54	0.52	0.89	1.00	0.86	0.53	0.69
concave points_worst	0.74	0.30	0.77	0.72	0.50	0.82	0.86	0.91	0.43	0.18	0.53	-0.12	0.55	0.54	-0.10	0.48	0.44	0.60	-0.03	0.22	0.79	0.36	0.82	0.75	0.55	0.80	0.86	1.00	0.50	0.51
symmetry_worst	0.16	0.11	0.19	0.14	0.39	0.51	0.41	0.38	0.70	0.33	0.09	-0.13	0.11	0.07	-0.11	0.28	0.20	0.14	0.39	0.11	0.24	0.23	0.27	0.21	0.49	0.61	0.53	0.50	1.00	0.54
fractal_dimension_worst	0.01	0.12	0.05	0.00	0.50	0.69	0.51	0.37	0.44	0.77	0.05	-0.05	0.09	0.02	0.10	0.59	0.44	0.31	0.08	0.59	0.09	0.22	0.14	0.08	0.62	0.81	0.69	0.51	0.54	1.00

0.4
0.2
0.0
−0.2

ENG 00:31

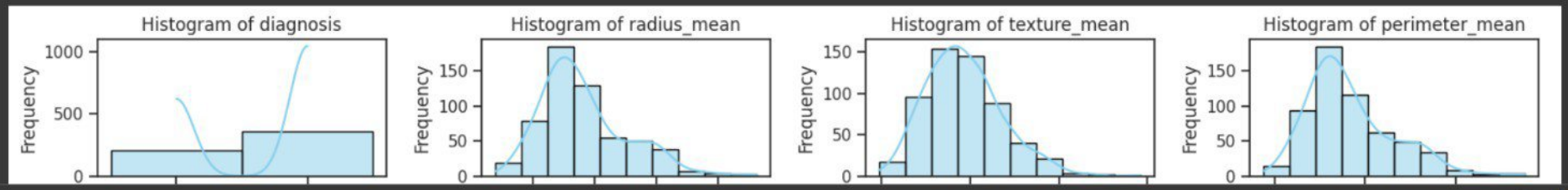
colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
[ ] import matplotlib.pyplot as plt
    import seaborn as sns
    import plotly.graph_objects as go
    import plotly.express as px
```

```
fig,axes = plt.subplots(nrows=8,ncols=4,figsize=(15,18))
column_names=df.columns
for i,column in enumerate(column_names):
    row = i // 4  # Calculate row index
    col = i % 4
    sns.histplot(data=df[column], bins=10, color='skyblue',kde=True, edgecolor='black', ax=axes[row, col])
    # Set labels and title
    axes[row, col].set_xlabel(f'{column}')
    axes[row, col].set_ylabel('Frequency')
    axes[row, col].set_title(f'Histogram of {column}')
# Adjust layout
plt.tight_layout()
# Show the plots
plt.show()
```



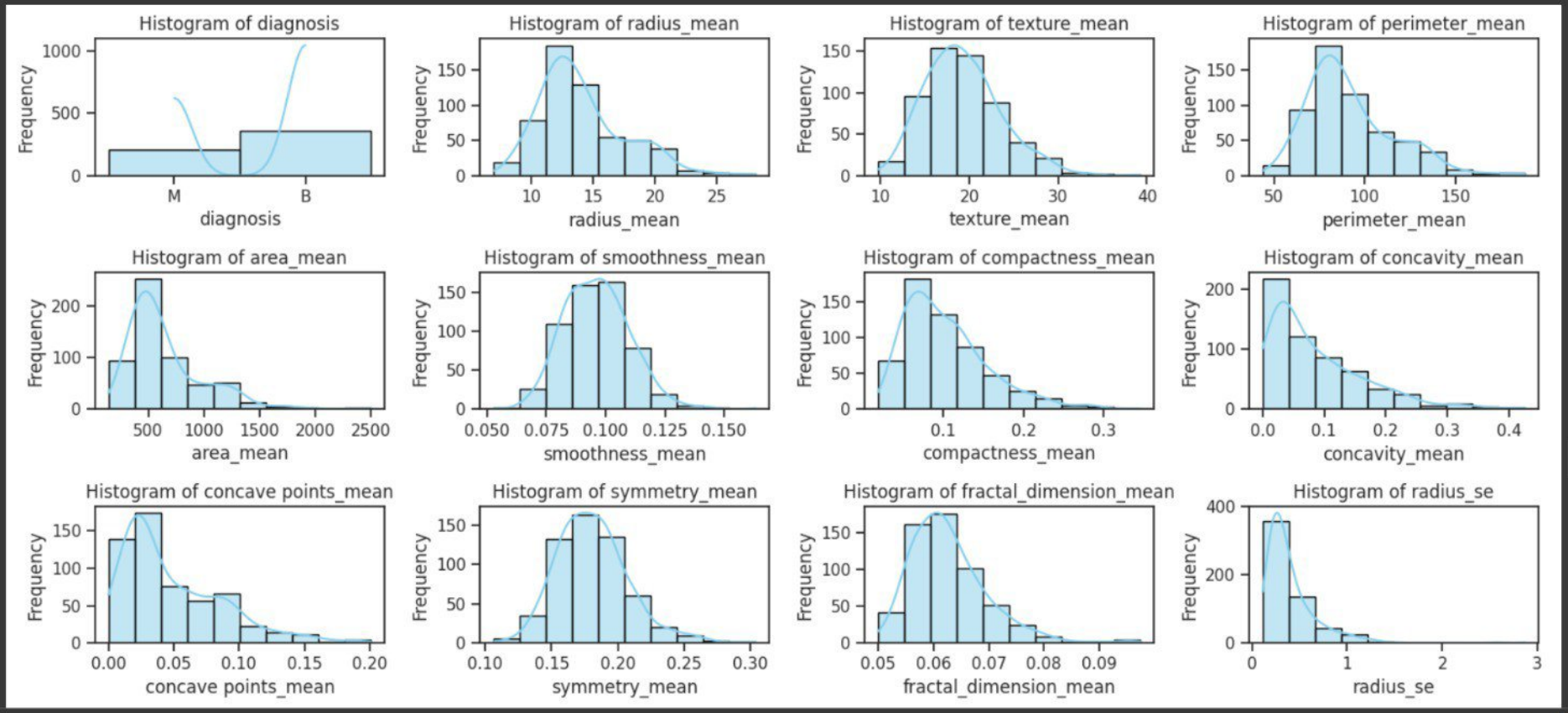
ENG 00:31

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

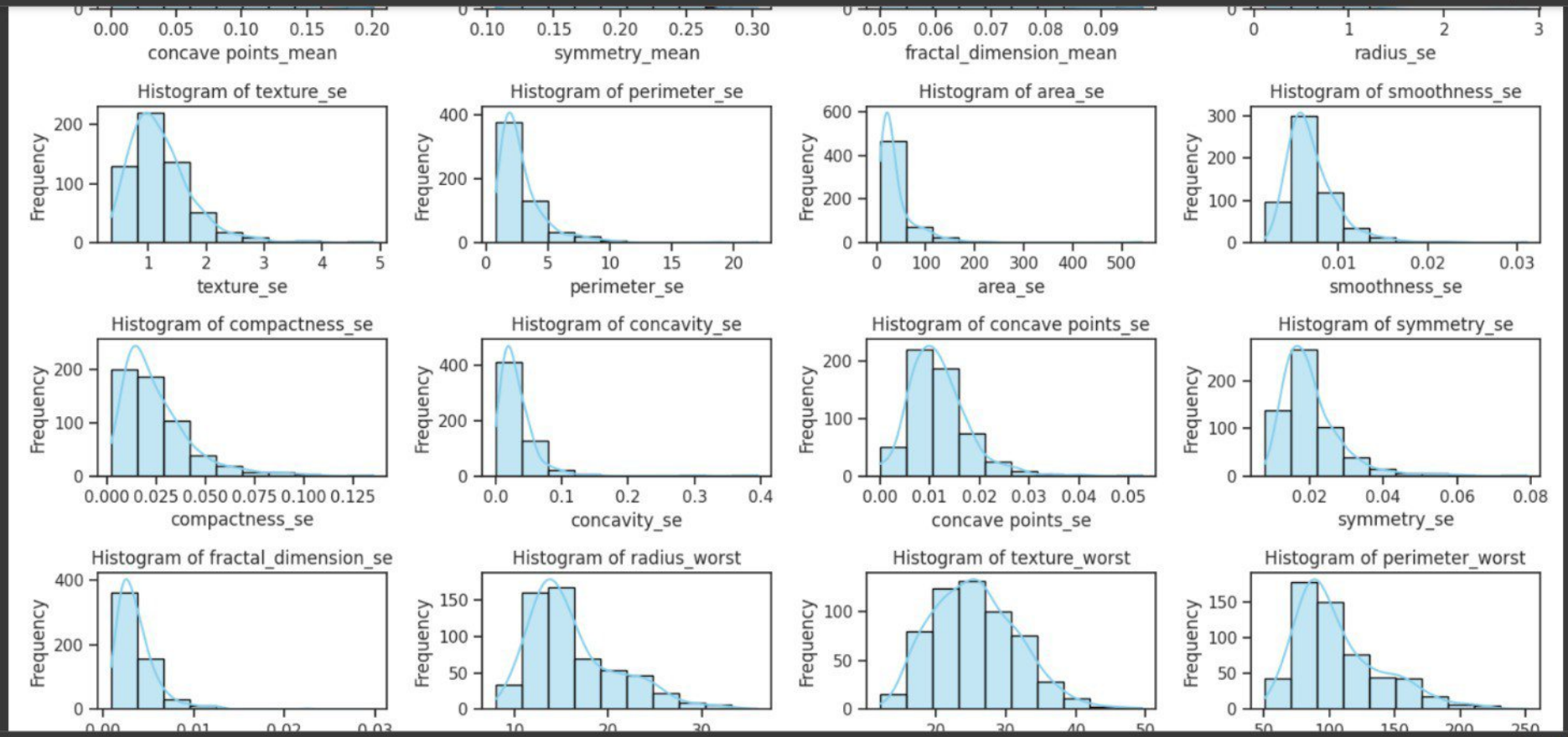
+ Code + Text

Connect Colab AI

[ ]



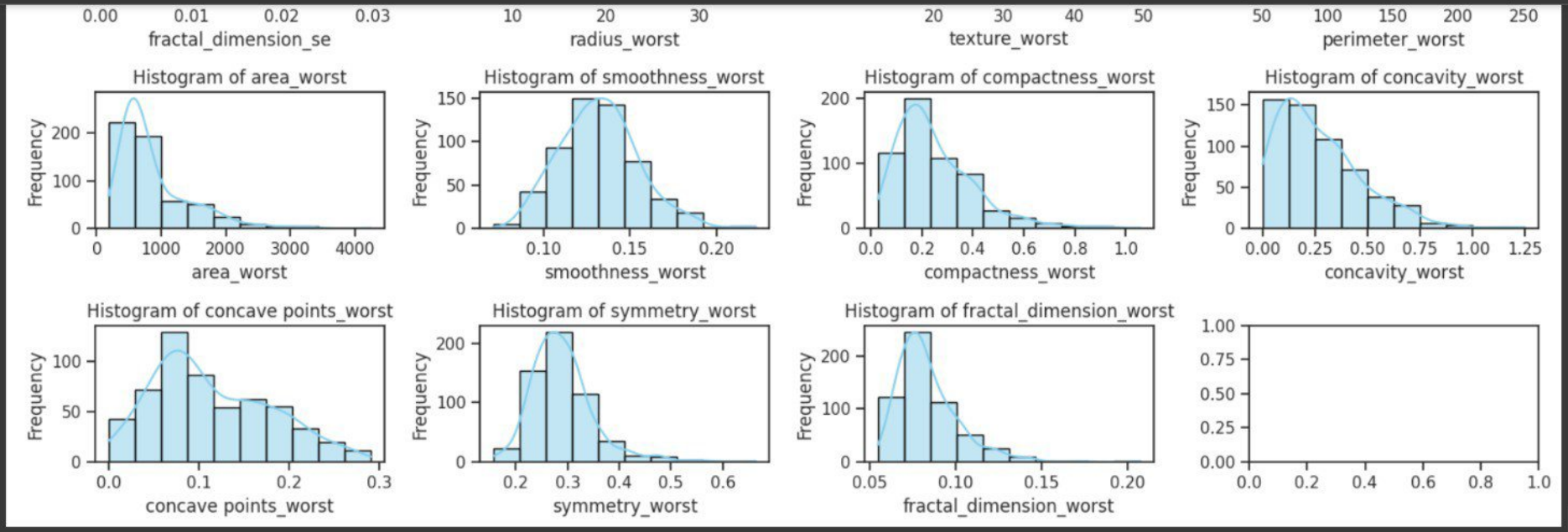
ENG 00:32


Histogram of texture_se
Histogram of perimeter_se
Histogram of area_se
Histogram of smoothness_se
Histogram of compactness_se
Histogram of concavity_se
Histogram of concave points_se
Histogram of symmetry_se
Histogram of fractal_dimension_se
Histogram of radius_worst
Histogram of texture_worst
Histogram of perimeter_worst
Frequency
texture_se
perimeter_se
area_se
smoothness_se
compactness_se
concavity_se
concave points_se
symmetry_se
concave points_mean
symmetry_mean
fractal_dimension_mean
radius_se

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI




```
[ ] x=df.drop(['diagnosis'],axis=1)
    x
```

	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fractal_dimen
0	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	

ENG 00:32

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
x=df.drop(['diagnosis'],axis=1)
x
```

	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fractal_dimen
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710	0.2419	
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017	0.1812	
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790	0.2069	
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520	0.2597	
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.19800	0.10430	0.1809	
...	...	...	...	...	...	...	...	...	...	
564	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	0.1726	
565	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791	0.1752	
566	16.60	28.08	108.30	858.1	0.08455	0.10230	0.09251	0.05302	0.1590	
567	20.60	29.33	140.10	1265.0	0.11780	0.27700	0.35140	0.15200	0.2397	
568	7.76	24.54	47.92	181.0	0.05263	0.04362	0.00000	0.00000	0.1587	

569 rows × 30 columns

ENG 00:32

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
y=df['diagnosis']
y
```

```
0      M
1      M
2      M
3      M
4      M
      ..
564    M
565    M
566    M
567    M
568    B
Name: diagnosis, Length: 569, dtype: object
```

```
[ ] from sklearn.model_selection import train_test_split
    x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.30,random_state=42)
    x_train
```

	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fractal_dimen
149	13.740	17.91	88.12	585.0	0.07944	0.06376	0.02881	0.01329	0.1473	
124	13.370	16.39	86.10	553.5	0.07115	0.07325	0.08092	0.02800	0.1422	
421	14.690	13.98	98.22	656.1	0.10310	0.18360	0.14500	0.06300	0.2086	

ENG 00:32

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1lh9v5G9TCoH

+ Code + Text

Connect Colab AI

```
x_test
```

	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fractal_dimen
204	12.470	18.60	81.09	481.9	0.09965	0.10580	0.08005	0.03821	0.1925	
70	18.940	21.31	123.60	1130.0	0.09009	0.10290	0.10800	0.07951	0.1582	
131	15.460	19.48	101.70	748.9	0.10920	0.12230	0.14660	0.08087	0.1931	
431	12.400	17.68	81.47	467.8	0.10540	0.13160	0.07741	0.02799	0.1811	
540	11.540	14.44	74.65	402.9	0.09984	0.11200	0.06737	0.02594	0.1818	
...	...	...	...	...	...	...	...	...	...	
69	12.780	16.49	81.37	502.5	0.09831	0.05234	0.03653	0.02864	0.1590	
542	14.740	25.42	94.70	668.6	0.08275	0.07214	0.04105	0.03027	0.1840	
176	9.904	18.06	64.60	302.4	0.09699	0.12940	0.13070	0.03716	0.1669	
501	13.820	24.49	92.33	595.9	0.11620	0.16810	0.13570	0.06759	0.2275	
247	12.890	14.11	84.95	512.2	0.08760	0.13460	0.13740	0.03980	0.1596	

171 rows × 30 columns

ENG 00:32

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
[ ] from sklearn.model_selection import train_test_split
    x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.30,random_state=42)
    x_train
```

	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fractal_dimen
149	13.740	17.91	88.12	585.0	0.07944	0.06376	0.02881	0.01329	0.1473	
124	13.370	16.39	86.10	553.5	0.07115	0.07325	0.08092	0.02800	0.1422	
421	14.690	13.98	98.22	656.1	0.10310	0.18360	0.14500	0.06300	0.2086	
195	12.910	16.33	82.53	516.4	0.07941	0.05366	0.03873	0.02377	0.1829	
545	13.620	23.23	87.19	573.2	0.09246	0.06747	0.02974	0.02443	0.1664	
...	...	...	...	...	...	...	...	...	...	
71	8.888	14.64	58.79	244.0	0.09783	0.15310	0.08606	0.02872	0.1902	
106	11.640	18.33	75.17	412.5	0.11420	0.10170	0.07070	0.03485	0.1801	
270	14.290	16.82	90.30	632.6	0.06429	0.02675	0.00725	0.00625	0.1508	
435	13.980	19.62	91.12	599.5	0.10600	0.11330	0.11260	0.06463	0.1669	
102	12.180	20.52	77.22	458.7	0.08013	0.04038	0.02383	0.01770	0.1739	

398 rows × 30 columns

ENG 00:32

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1lh9v5G9TCoH

+ Code + Text

Connect Colab AI

```
y_train
```

```
149    B
124    B
421    B
195    B
545    B
      ..
71     B
106    B
270    B
435    M
102    B
Name: diagnosis, Length: 398, dtype: object
```

```
y_test
```

```
204    B
70     M
131    M
431    B
540    B
      ..
69     B
542    B
176    B
501    M
247    B
Name: diagnosis, Length: 171, dtype: object
```

ENG 00:32

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
[ ] for i in lst_model:
      print('model is',i)
      i.fit(x_train,y_train)
      y_pred=i.predict(x_test)
      print("*"*100)
      print(confusion_matrix(y_test,y_pred))
      print("Accuracy score is",accuracy_score(y_test,y_pred))
      print(".......classification Report........")
      print(classification_report(y_test,y_pred))
```

```
model is KNeighborsClassifier(n_neighbors=7)
****************************************************************************************************
[[106   2]
 [  4  59]]
Accuracy score is 0.9649122807017544
.......classification Report........
              precision    recall  f1-score   support

           B       0.96      0.98      0.97       108
           M       0.97      0.94      0.95        63

    accuracy                           0.96       171
   macro avg       0.97      0.96      0.96       171
weighted avg       0.96      0.96      0.96       171

model is GaussianNB()
****************************************************************************************************
[[104   4]
 [  6  57]]
```

ENG 00:32

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
[ ] from sklearn.neighbors import KNeighborsClassifier
    from sklearn.naive_bayes import GaussianNB
    from sklearn.svm import SVC
    from sklearn.ensemble import RandomForestClassifier
    from sklearn.tree import DecisionTreeClassifier
    from sklearn.metrics import confusion_matrix,accuracy_score
    from sklearn.metrics import classification_report
    k_model=KNeighborsClassifier(n_neighbors=7)
    n_model=GaussianNB()
    s_model=SVC()
    r_model=RandomForestClassifier()
    d_model=DecisionTreeClassifier(criterion='entropy')
    lst_model=[k_model,n_model,s_model,r_model,d_model]
```

```
[ ] for i in lst_model:
      print('model is',i)
      i.fit(x_train,y_train)
      y_pred=i.predict(x_test)
      print("*"*100)
      print(confusion_matrix(y_test,y_pred))
      print("Accuracy score is",accuracy_score(y_test,y_pred))
      print(".......classification Report........")
      print(classification_report(y_test,y_pred))
```

```
model is KNeighborsClassifier(n_neighbors=7)
****************************************************************************************************
[[106   2]
 [  4  59]]
```

ENG 00:32

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
[ ] model is KNeighborsClassifier(n_neighbors=7)
    ****************************************************************************************
    [[106   2]
     [  4  59]]
    Accuracy score is 0.9649122807017544
    .......classification Report........
                  precision    recall  f1-score   support

               B       0.96      0.98      0.97       108
               M       0.97      0.94      0.95        63

        accuracy                           0.96       171
       macro avg       0.97      0.96      0.96       171
    weighted avg       0.96      0.96      0.96       171

    model is GaussianNB()
    ****************************************************************************************
    [[104   4]
     [  6  57]]
    Accuracy score is 0.9415204678362573
    .......classification Report........
                  precision    recall  f1-score   support

               B       0.95      0.96      0.95       108
               M       0.93      0.90      0.92        63

        accuracy                           0.94       171
       macro avg       0.94      0.93      0.94       171
    weighted avg       0.94      0.94      0.94       171

    model is SVC()
```

ENG 00:32

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
[ ] model is SVC()
    ****************************************************************************************
    [[108   0]
     [ 11  52]]
    Accuracy score is 0.935672514619883
    .......classification Report........
                  precision    recall  f1-score   support

               B       0.91      1.00      0.95       108
               M       1.00      0.83      0.90        63

        accuracy                           0.94       171
       macro avg       0.95      0.91      0.93       171
    weighted avg       0.94      0.94      0.93       171

    model is RandomForestClassifier()
    ****************************************************************************************
    [[106   2]
     [  4  59]]
    Accuracy score is 0.9649122807017544
    .......classification Report........
                  precision    recall  f1-score   support

               B       0.96      0.98      0.97       108
               M       0.97      0.94      0.95        63

        accuracy                           0.96       171
       macro avg       0.97      0.96      0.96       171
    weighted avg       0.96      0.96      0.96       171
```

ENG 00:32

colab.research.google.com/drive/1JXDKQi_10-lB8J3ArI_Cp9SPDypOswiH#scrollTo=1Ih9v5G9TCoH

+ Code + Text

Connect Colab AI

```
[ ] model is RandomForestClassifier()
    ****************************************************************************************
    [[106   2]
     [  4  59]]
    Accuracy score is 0.9649122807017544
    .......classification Report........
                  precision    recall  f1-score   support

               B       0.96      0.98      0.97       108
               M       0.97      0.94      0.95        63

        accuracy                           0.96       171
       macro avg       0.97      0.96      0.96       171
    weighted avg       0.96      0.96      0.96       171

    model is DecisionTreeClassifier(criterion='entropy')
    ****************************************************************************************
    [[106   2]
     [  4  59]]
    Accuracy score is 0.9649122807017544
    .......classification Report........
                  precision    recall  f1-score   support

               B       0.96      0.98      0.97       108
               M       0.97      0.94      0.95        63

        accuracy                           0.96       171
       macro avg       0.97      0.96      0.96       171
    weighted avg       0.96      0.96      0.96       171
```

ENG 00:32