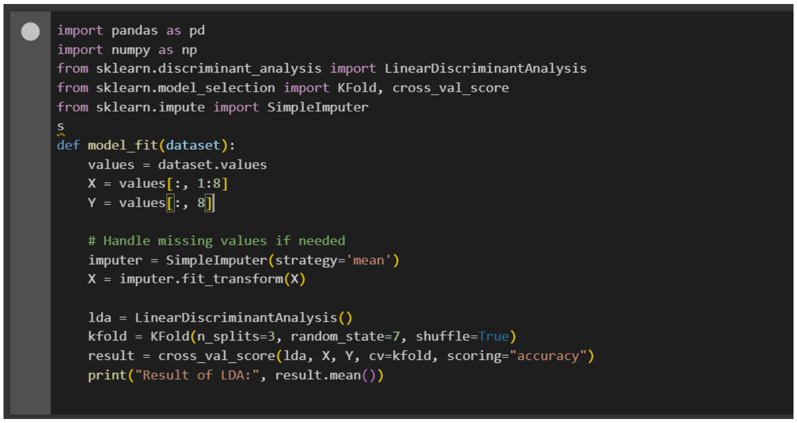
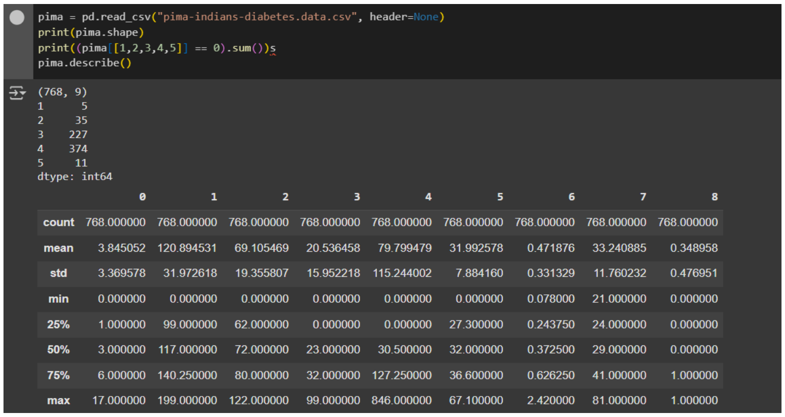
Name: Durva Kadam   Roll No: 11

Subject: AI-ML in Healthcare Lab

Experiment – 1: To collect, clean, integrate and transform Healthcare data based on specific disease.



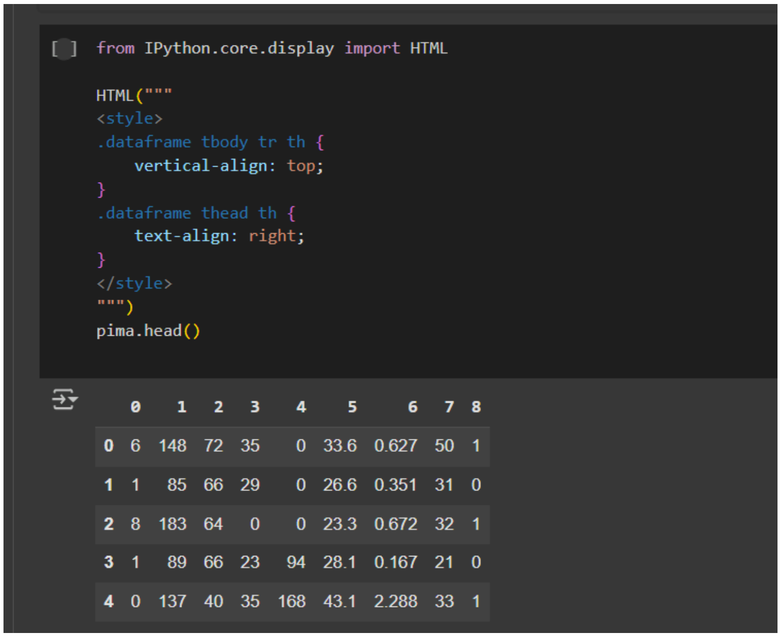
```
import pandas as pd
import numpy as np
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.model_selection import KFold, cross_val_score
from sklearn.impute import SimpleImputer
s
def model_fit(dataset):
    values = dataset.values
    X = values[:, 1:8]
    Y = values[:, 8]

    # Handle missing values if needed
    imputer = SimpleImputer(strategy='mean')
    X = imputer.fit_transform(X)

    lda = LinearDiscriminantAnalysis()
    kfold = KFold(n_splits=3, random_state=7, shuffle=True)
    result = cross_val_score(lda, X, Y, cv=kfold, scoring="accuracy")
    print("Result of LDA:", result.mean())
```



```
pima = pd.read_csv("pima-indians-diabetes.data.csv", header=None)
print(pima.shape)
print((pima[[1,2,3,4,5]] == 0).sum())s
pima.describe()
```

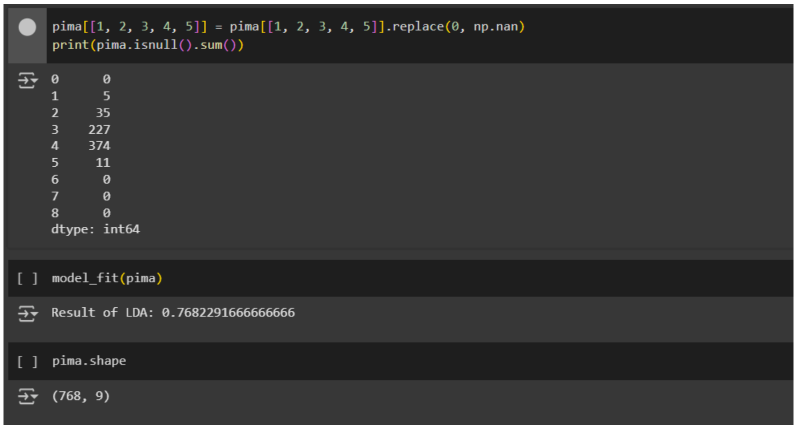
```
(768, 9)
1      5
2     35
3    227
4    374
5     11
dtype: int64
```

	0	1	2	3	4	5	6	7	8
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	0.471876	33.240885	0.348958
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	0.331329	11.760232	0.476951
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000	21.000000	0.000000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.243750	24.000000	0.000000
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	0.372500	29.000000	0.000000
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	0.626250	41.000000	1.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	81.000000	1.000000



```
from IPython.core.display import HTML

HTML("""
<style>
.dataframe tbody tr th {
    vertical-align: top;
}
.dataframe thead th {
    text-align: right;
}
</style>
""")
pima.head()
```

	0	1	2	3	4	5	6	7	8
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1



```
pima[[1, 2, 3, 4, 5]] = pima[[1, 2, 3, 4, 5]].replace(0, np.nan)
print(pima.isnull().sum())
```

```
0      0
1      5
2     35
3    227
4    374
5     11
6      0
7      0
8      0
dtype: int64
```

```
[ ] model_fit(pima)
```

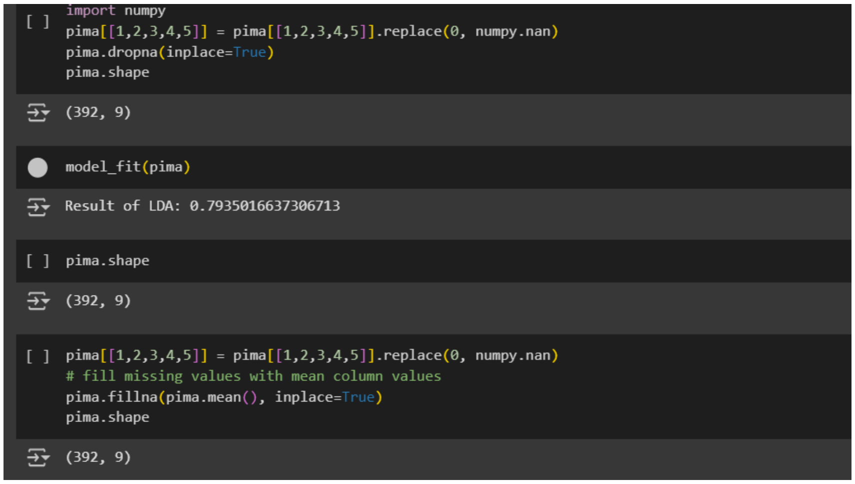
```
Result of LDA: 0.7682291666666666
```

```
[ ] pima.shape
```

```
(768, 9)
```

Name: Durva Kadam

Roll No: 11



```
[ ] import numpy
    pima[[1,2,3,4,5]] = pima[[1,2,3,4,5]].replace(0, numpy.nan)
    pima.dropna(inplace=True)
    pima.shape
```

```
(392, 9)
```

```
model_fit(pima)
```

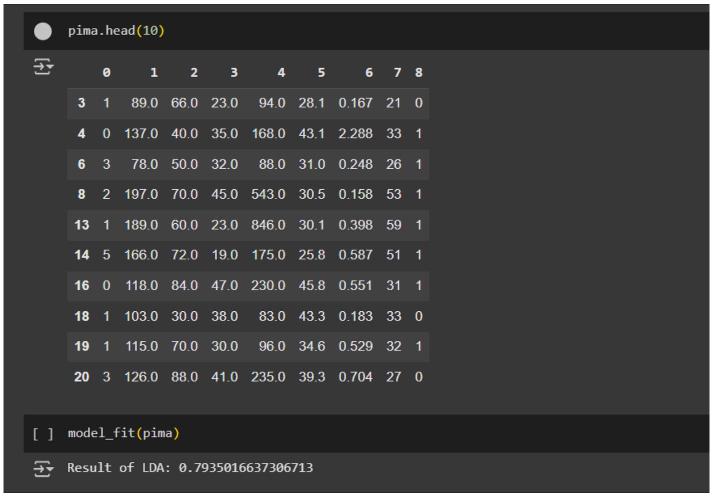
```
Result of LDA: 0.7935016637306713
```

```
[ ] pima.shape
```

```
(392, 9)
```

```
[ ] pima[[1,2,3,4,5]] = pima[[1,2,3,4,5]].replace(0, numpy.nan)
    # fill missing values with mean column values
    pima.fillna(pima.mean(), inplace=True)
    pima.shape
```

```
(392, 9)
```

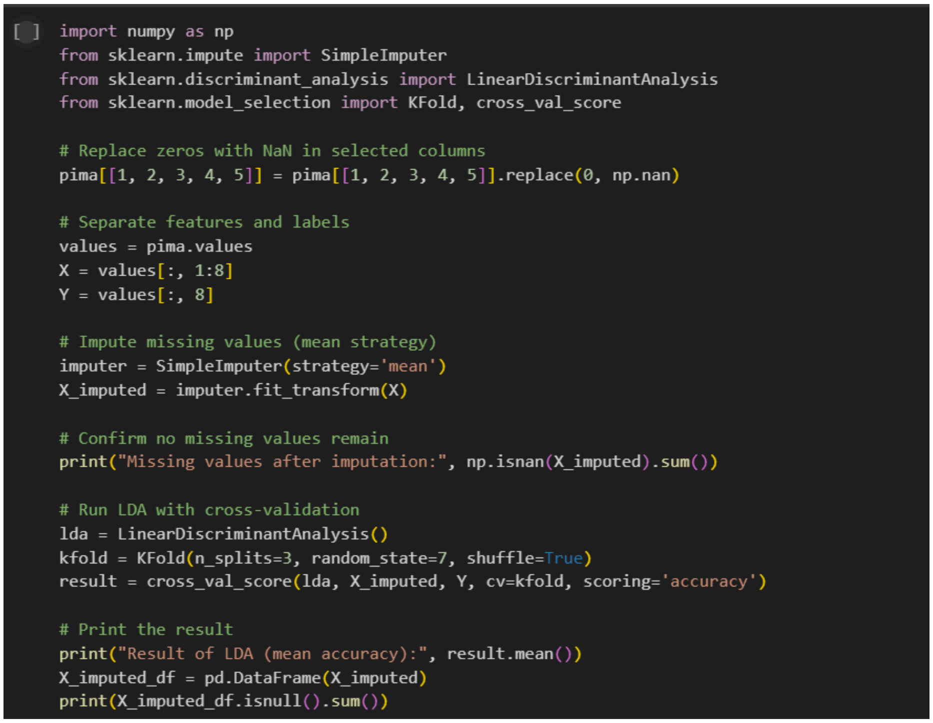


```
pima.head(10)
```

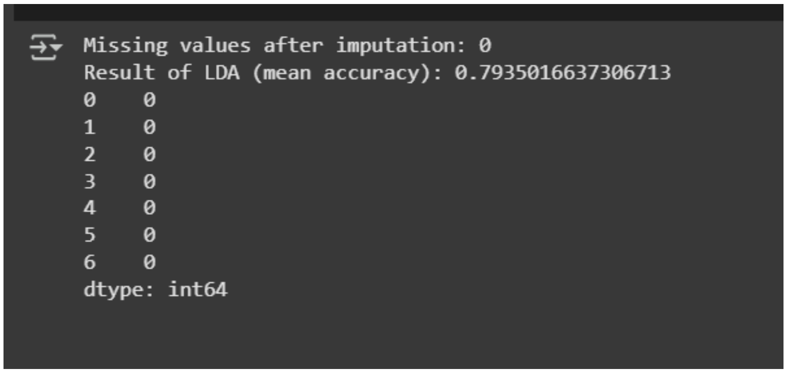
	0	1	2	3	4	5	6	7	8
3	1	89.0	66.0	23.0	94.0	28.1	0.167	21	0
4	0	137.0	40.0	35.0	168.0	43.1	2.288	33	1
6	3	78.0	50.0	32.0	88.0	31.0	0.248	26	1
8	2	197.0	70.0	45.0	543.0	30.5	0.158	53	1
13	1	189.0	60.0	23.0	846.0	30.1	0.398	59	1
14	5	166.0	72.0	19.0	175.0	25.8	0.587	51	1
16	0	118.0	84.0	47.0	230.0	45.8	0.551	31	1
18	1	103.0	30.0	38.0	83.0	43.3	0.183	33	0
19	1	115.0	70.0	30.0	96.0	34.6	0.529	32	1
20	3	126.0	88.0	41.0	235.0	39.3	0.704	27	0

```
[ ] model_fit(pima)
```

```
Result of LDA: 0.7935016637306713
```



```
[ ] import numpy as np
    from sklearn.impute import SimpleImputer
    from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
    from sklearn.model_selection import KFold, cross_val_score

    # Replace zeros with NaN in selected columns
    pima[[1, 2, 3, 4, 5]] = pima[[1, 2, 3, 4, 5]].replace(0, np.nan)

    # Separate features and labels
    values = pima.values
    X = values[:, 1:8]
    Y = values[:, 8]

    # Impute missing values (mean strategy)
    imputer = SimpleImputer(strategy='mean')
    X_imputed = imputer.fit_transform(X)

    # Confirm no missing values remain
    print("Missing values after imputation:", np.isnan(X_imputed).sum())

    # Run LDA with cross-validation
    lda = LinearDiscriminantAnalysis()
    kfold = KFold(n_splits=3, random_state=7, shuffle=True)
    result = cross_val_score(lda, X_imputed, Y, cv=kfold, scoring='accuracy')

    # Print the result
    print("Result of LDA (mean accuracy):", result.mean())
    X_imputed_df = pd.DataFrame(X_imputed)
    print(X_imputed_df.isnull().sum())
```



```
Missing values after imputation: 0
Result of LDA (mean accuracy): 0.7935016637306713
0    0
1    0
2    0
3    0
4    0
5    0
6    0
dtype: int64
```