Experiment No. 7
Apply Dimensionality Reduction on Adult Census Income Dataset and analyze the performance of the model
Date of Performance: 27-09-2023
Date of Submission: 09-10-2023

Aim: Apply Dimensionality Reduction on Adult Census Income Dataset and analyze the performance of the model.

Objective: Able to perform various feature engineering tasks, perform dimetionality reduction on the given dataset and maximize the accuracy, Precision, Recall, F1 score.

Theory:

In machine learning classification problems, there are often too many factors on the basis of which the final classification is done. These factors are basically variables called features. The higher the number of features, the harder it gets to visualize the training set and then work on it. Sometimes, most of these features are correlated, and hence redundant. This is where dimensionality reduction algorithms come into play. Dimensionality reduction is the process of reducing the number of random variables under consideration, by obtaining a set of principal variables. It can be divided into feature selection and feature extraction.

Dataset:

Predict whether income exceeds $50K/yr based on census data. Also known as "Adult" dataset.

Attribute Information:

Listing of attributes:

>50K, <=50K.

age: continuous.

workclass: Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.

fnlwgt: continuous.

education: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.

education-num: continuous.

marital-status: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.

occupation: Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces.

relationship: Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.

race: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.

sex: Female, Male.

capital-gain: continuous.

capital-loss: continuous.

hours-per-week: continuous.

native-country: United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad & Tobago, Peru, Hong, Holand-Netherlands.

Code:

Dimensionality Reduction

Adult Census Income Dataset

[420]:
```
import pandas as pd
import numpy as np
import seaborn as sb
from sklearn.model_selection import train_test_split
from  sklearn.metrics import accuracy_score
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import GradientBoostingClassifier
import matplotlib.pyplot as plt
```

[421]:
```
df=pd.read_csv("/content/adult.csv",error_bad_lines=False, engine="python")
```

```
<ipython-input-421-8dbf926b9719>:1: FutureWarning: The error_bad_lines argument
has been deprecated and will be removed in a future version. Use on_bad_lines in
the future.


  df=pd.read_csv("/content/adult.csv",error_bad_lines=False, engine="python")
```

[422]:
```
print(df)
```

```
       age workclass  fnlwgt     education  education.num      marital.status  \
0       90         ?   77053       HS-grad              9             Widowed
1       82   Private  132870       HS-grad              9             Widowed
2       66         ?  186061  Some-college             10             Widowed
3       54   Private  140359       7th-8th              4            Divorced
4       41   Private  264663  Some-college             10           Separated
…     …         …     …            …            …                 …
32556   22   Private  310152  Some-college             10       Never-married
32557   27   Private  257302    Assoc-acdm             12  Married-civ-spouse
32558   40   Private  154374       HS-grad              9  Married-civ-spouse
32559   58   Private  151910       HS-grad              9             Widowed
32560   22   Private  201490       HS-grad              9       Never-married

              occupation   relationship   race     sex  capital.gain  \
0                      ?  Not-in-family  White  Female             0
1        Exec-managerial  Not-in-family  White  Female             0
2                      ?      Unmarried  Black  Female             0
```

```
3      Machine-op-inspct      Unmarried  White  Female             0
4         Prof-specialty      Own-child  White  Female             0
…                    …              …     …      …           …
32556    Protective-serv  Not-in-family  White    Male             0
32557       Tech-support           Wife  White  Female             0
32558  Machine-op-inspct        Husband  White    Male             0
32559       Adm-clerical      Unmarried  White  Female             0
32560       Adm-clerical      Own-child  White    Male             0

       capital.loss  hours.per.week native.country income
0              4356              40  United-States  <=50K
1              4356              18  United-States  <=50K
2              4356              40  United-States  <=50K
3              3900              40  United-States  <=50K
4              3900              40  United-States  <=50K
…                 …               …            …    …
32556             0              40  United-States  <=50K
32557             0              38  United-States  <=50K
32558             0              40  United-States   >50K
32559             0              40  United-States  <=50K
32560             0              20  United-States  <=50K

[32561 rows x 15 columns]
```

[423]:
```python
df.describe
```

[423]:
```
<bound method NDFrame.describe of         age workclass  fnlwgt     education
education.num      marital.status  \
0       90         ?   77053       HS-grad              9             Widowed
1       82   Private  132870       HS-grad              9             Widowed
2       66         ?  186061  Some-college             10             Widowed
3       54   Private  140359       7th-8th              4            Divorced
4       41   Private  264663  Some-college             10           Separated
…     …        …     …           …              …                   …
32556   22   Private  310152  Some-college             10       Never-married
32557   27   Private  257302    Assoc-acdm             12  Married-civ-spouse
32558   40   Private  154374       HS-grad              9  Married-civ-spouse
32559   58   Private  151910       HS-grad              9             Widowed
32560   22   Private  201490       HS-grad              9       Never-married

              occupation   relationship   race     sex  capital.gain  \
0                      ?  Not-in-family  White  Female             0
1        Exec-managerial  Not-in-family  White  Female             0
2                      ?      Unmarried  Black  Female             0
3      Machine-op-inspct      Unmarried  White  Female             0
4         Prof-specialty      Own-child  White  Female             0
…                    …              …     …      …           …
```

```
32556    Protective-serv  Not-in-family  White    Male             0
32557       Tech-support           Wife  White  Female             0
32558  Machine-op-inspct        Husband  White    Male             0
32559       Adm-clerical      Unmarried  White  Female             0
32560       Adm-clerical      Own-child  White    Male             0

       capital.loss  hours.per.week native.country income
0              4356              40  United-States  <=50K
1              4356              18  United-States  <=50K
2              4356              40  United-States  <=50K
3              3900              40  United-States  <=50K
4              3900              40  United-States  <=50K
…                 …               …              …    …
32556             0              40  United-States  <=50K
32557             0              38  United-States  <=50K
32558             0              40  United-States   >50K
32559             0              40  United-States  <=50K
32560             0              20  United-States  <=50K

[32561 rows x 15 columns]>
```

[424]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32561 entries, 0 to 32560
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   age             32561 non-null  int64
 1   workclass       32561 non-null  object
 2   fnlwgt          32561 non-null  int64
 3   education       32561 non-null  object
 4   education.num   32561 non-null  int64
 5   marital.status  32561 non-null  object
 6   occupation      32561 non-null  object
 7   relationship    32561 non-null  object
 8   race            32561 non-null  object
 9   sex             32561 non-null  object
 10  capital.gain    32561 non-null  int64
 11  capital.loss    32561 non-null  int64
 12  hours.per.week  32561 non-null  int64
 13  native.country  32561 non-null  object
 14  income          32561 non-null  object
dtypes: int64(6), object(9)
memory usage: 3.7+ MB
```

[425]:
```
df[df == '?'] = np.nan
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32561 entries, 0 to 32560
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   age             32561 non-null  int64
 1   workclass       30725 non-null  object
 2   fnlwgt          32561 non-null  int64
 3   education       32561 non-null  object
 4   education.num   32561 non-null  int64
 5   marital.status  32561 non-null  object
 6   occupation      30718 non-null  object
 7   relationship    32561 non-null  object
 8   race            32561 non-null  object
 9   sex             32561 non-null  object
 10  capital.gain    32561 non-null  int64
 11  capital.loss    32561 non-null  int64
 12  hours.per.week  32561 non-null  int64
 13  native.country  31978 non-null  object
 14  income          32561 non-null  object
dtypes: int64(6), object(9)
memory usage: 3.7+ MB
```

[426]:
```
df.isnull().sum()
```

[426]:
```
age                  0
workclass         1836
fnlwgt               0
education            0
education.num        0
marital.status       0
occupation        1843
relationship         0
race                 0
sex                  0
capital.gain         0
capital.loss         0
hours.per.week       0
native.country     583
income               0
dtype: int64
```

[427]:
```
max_category = df['workclass'].value_counts().idxmax()
df['workclass'].fillna(max_category, inplace=True)
```

```
max_category = df['occupation'].value_counts().idxmax()
df['occupation'].fillna(max_category, inplace=True)
max_category = df['native.country'].value_counts().idxmax()
df['native.country'].fillna(max_category, inplace=True)
```

[428]:
```
df.isnull().sum()
```

[428]:
```
age               0
workclass         0
fnlwgt            0
education         0
education.num     0
marital.status    0
occupation        0
relationship      0
race              0
sex               0
capital.gain      0
capital.loss      0
hours.per.week    0
native.country    0
income            0
dtype: int64
```

[429]:
```
from sklearn.preprocessing import LabelEncoder
```

[430]:
```
labelencoder_x=LabelEncoder()
df["workclass"]=labelencoder_x.fit_transform(df["workclass"])
df["education"]=labelencoder_x.fit_transform(df["education"])
df["relationship"]=labelencoder_x.fit_transform(df["relationship"])
df["occupation"]=labelencoder_x.fit_transform(df["occupation"])
df["sex"]=labelencoder_x.fit_transform(df["sex"])
df["income"]=labelencoder_x.fit_transform(df["income"])
df["marital.status"]=labelencoder_x.fit_transform(df["marital.status"])
df["race"]=labelencoder_x.fit_transform(df["race"])
df["native.country"]=labelencoder_x.fit_transform(df["native.country"])
```

[431]:
```
x=df.drop("income",axis=1)
y=df["income"]
```

[432]:
```
df.head(20)
```

[432]:
```
   age  workclass  fnlwgt  education  education.num  marital.status  \
0   90          3   77053         11              9               6
1   82          3  132870         11              9               6
2   66          3  186061         15             10               6
3   54          3  140359          5              4               0
```

4	41	3	264663	15	10	5
5	34	3	216864	11	9	0
6	38	3	150601	0	6	5
7	74	6	88638	10	16	4
8	68	0	422013	11	9	0
9	41	3	70037	15	10	4
10	45	3	172274	10	16	0
11	38	5	164526	14	15	4
12	52	3	129177	9	13	6
13	32	3	136204	12	14	5
14	51	3	172175	10	16	4
15	46	3	45363	14	15	0
16	45	3	172822	1	7	0
17	57	3	317847	12	14	0
18	22	3	119592	7	12	4
19	34	3	203034	9	13	5

	occupation	relationship	race	sex	capital.gain	capital.loss \
0	9	1	4	0	0	4356
1	3	1	4	0	0	4356
2	9	4	2	0	0	4356
3	6	4	4	0	0	3900
4	9	3	4	0	0	3900
5	7	4	4	0	0	3770
6	0	4	4	1	0	3770
7	9	2	4	0	0	3683
8	9	1	4	0	0	3683
9	2	4	4	1	0	3004
10	9	4	2	0	0	3004
11	9	1	4	1	0	2824
12	7	1	4	0	0	2824
13	3	1	4	1	0	2824
14	9	1	4	1	0	2824
15	9	1	4	1	0	2824
16	13	1	4	1	0	2824
17	3	1	4	1	0	2824
18	5	1	2	1	0	2824
19	11	1	4	1	0	2824

	hours.per.week	native.country	income
0	40	38	0
1	18	38	0
2	40	38	0
3	40	38	0
4	40	38	0
5	45	38	0
6	40	38	0

7	20	38	1
8	40	38	0
9	60	38	1
10	35	38	1
11	45	38	1
12	20	38	1
13	55	38	1
14	40	38	1
15	40	38	1
16	76	38	1
17	50	38	1
18	40	38	1
19	50	38	1

[433]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=0,test_size=0.3)
```

[434]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

[435]:
```
LR = LogisticRegression()
LR.fit(x_train,y_train)
```

[435]: LogisticRegression()

[436]:
```
y_pred = LR.predict(x_test)
```

[437]:
```
accuracy_score(y_test, y_pred)
```

[437]: 0.7923021803664654

[438]:
```
from sklearn.decomposition import PCA
pca=PCA()
```

[439]:
```
x_train = pca.fit_transform(x_train)
pca.explained_variance_ratio_
```

[439]:
```
array([9.95218448e-01, 4.76707843e-03, 1.44365982e-05, 1.65490513e-08,
       1.31157437e-08, 3.26144185e-09, 1.44104367e-09, 1.37401932e-09,
       4.46008924e-10, 2.13186968e-10, 1.72160573e-10, 1.09384760e-10,
       6.12487477e-11, 1.22057563e-11])
```

[440]:
```
df=df.drop(columns='race')
df=df.drop(columns='fnlwgt')
df=df.drop(columns='education.num')
df=df.drop(columns='relationship')
df=df.drop(columns='native.country')
# df=df.drop(columns='marital.status')
```

[441]:
```
x = df.drop(['income'], axis=1)
y = df['income']
```

[442]:
```
pca= PCA()
pca.fit(x_train)
cumsum = np.cumsum(pca.explained_variance_ratio_)
dim = np.argmax(cumsum >= 0.90) + 1
print('The number of dimensions required to preserve 90% of variance is',dim)
```

```
The number of dimensions required to preserve 90% of variance is 1
```

[443]:
```
x = df.drop(['income'], axis=1)
y = df['income']
```

[444]:
```
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.3,␣
↪random_state = 0)
```

[445]:
```
LR2 = LogisticRegression()
LR2.fit(x_train, y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logistic.py:458:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  n_iter_i = _check_optimize_result(
```

[445]:
```
LogisticRegression()
```

[446]:
```
y_pred2 = LR2.predict(x_test)
```

[447]:
```
accuracy_score(y_test, y_pred2)
```

[447]:
```
0.7953731190500563
```

[448]:
```
from sklearn.metrics import confusion_matrix
import pandas as pd
```

[449]:
```
confusion = confusion_matrix(y_test, y_pred2)
```

[450]:
```
df_confusion = pd.DataFrame(confusion, columns=['Predicted No', 'Predicted␣
↪Yes'], index=['Actual No', 'Actual Yes'])
```

[451]:
```
from sklearn.metrics import classification_report
print(classification_report(y_test, y_pred))
```

```
              precision    recall  f1-score   support

           0       0.81      0.95      0.87      7410
           1       0.66      0.29      0.40      2359

    accuracy                           0.79      9769
   macro avg       0.73      0.62      0.64      9769
weighted avg       0.77      0.79      0.76      9769
```

Conclusion:

Prior to dimensionality reduction, the logistic regression model achieved an accuracy of approximately 0.7923. After dimensionality reduction, the model's accuracy improved to around 0.7953, indicating a slight performance gain.

When examining the >50K class, the precision was 0.66, the recall was 0.29, and the F1-score was 0.40. For the <=50K class, the precision was 0.81, the recall was 0.95, and the F1-score was 0.87.

Dimensionality reduction offers the benefit of simplifying the model by reducing feature complexity. This results in improved computational efficiency while still maintaining a reasonably high level of predictive performance.