EX-05-Feature-Generation

AIM

To read the given data and perform Feature Generation process and save the data to a file.

Explanation

Feature Generation (also known as feature construction, feature extraction or feature engineering) is the process of transforming features into new features that better relate to the target. It includes two process:

```
1.Feature Encoding
2.Feature Scaling
```

FEATURE ENCODING:

1. Ordinal Encoding An ordinal encoding involves mapping each unique label to an integer value. This type of encoding is really only appropriate if there is a known relationship between the categories. This relationship does exist for some of the variables in our dataset, and ideally, this should be harnessed when preparing the data.
2. Label Encoding Label encoding is a simple and straight forward approach. This converts each value in a categorical column into a numerical value. Each value in a categorical column is called Label.
3. Binary Encoding Binary encoding converts a category into binary digits. Each binary digit creates one feature column. If there are n unique categories, then binary encoding results in the only log(base 2)^n features.
4. One Hot Encoding We use this categorical data encoding technique when the features are nominal(do not have any order). In one hot encoding, for each level of a categorical feature, we create a new variable. Each category is mapped with a binary variable containing either 0 or 1. Here, 0 represents the absence, and 1 represents the presence of that category.

FEATURE SCALING:

1. Standard Scaler It is also called Z-score normalization. It calculates the z-score of each value and replaces the value with the calculated Z-score. The features are then rescaled with $\bar{x}$ =0 and σ=1
2. MinMaxScaler It is also referred to as Normalization. The features are scaled between 0 and 1. Here, the mean value remains same as in Standardization, that is, 0.
3. Maximum absolute scaling Maximum absolute scaling scales the data to its maximum value; that is, it divides every observation by the maximum value of the variable.The result of the preceding transformation is a distribution in which the values vary approximately within the range of -1 to 1.
4. RobustScaler RobustScaler transforms the feature vector by subtracting the median and then dividing by the interquartile range (75% value — 25% value).

ALGORITHM

STEP 1

Read the given Data

STEP 2

Clean the Data Set using Data Cleaning Process

STEP 3

Apply Feature Generation techniques to all the feature of the data set

STEP 4

Save the data to the file

CODE

Data.csv :

```
import pandas as pd
df=pd.read_csv("data.csv")
df

#feature generation
import category_encoders as ce
be=ce.BinaryEncoder()
ndf=be.fit_transform(df["bin_1"])
df["bin_1"] = be.fit_transform(df["bin_1"])
ndf

ndf2=be.fit_transform(df["bin_2"])
df["bin_2"] = be.fit_transform(df["bin_2"])
ndf2

df1=df.copy()
from sklearn.preprocessing import LabelEncoder,OrdinalEncoder,OneHotEncoder
import category_encoders as ce
be=ce.BinaryEncoder()
ohe=OneHotEncoder(sparse=False)
le=LabelEncoder()
oe=OrdinalEncoder()


df1["City"] = ohe.fit_transform(df1[["City"]])

temp=['Cold','Warm','Hot','Very Hot']
oe1=OrdinalEncoder(categories=[temp])
df1['Ord_1'] = oe1.fit_transform(df1[["Ord_1"]])

edu=['High School','Diploma','Bachelors','Masters','PhD']
oe2=OrdinalEncoder(categories=[edu])
df1['Ord_2']= oe2.fit_transform(df1[["Ord_2"]])
df1

#feature scaling
from sklearn.preprocessing import MinMaxScaler
sc=MinMaxScaler()
df2=pd.DataFrame(sc.fit_transform(df1),columns=['id', 'bin_1', 'bin_2', 'City', 'Ord_1','Ord_2','Target'])
df2

from sklearn.preprocessing import StandardScaler
sc1=StandardScaler()
df3=pd.DataFrame(sc1.fit_transform(df1),columns=['id', 'bin_1', 'bin_2', 'City', 'Ord_1','Ord_2','Target'])
df3

from sklearn.preprocessing import MaxAbsScaler
sc2=MaxAbsScaler()
df4=pd.DataFrame(sc2.fit_transform(df1),columns=['id', 'bin_1', 'bin_2', 'City', 'Ord_1','Ord_2','Target'])
```

```
df4

from sklearn.preprocessing import RobustScaler
sc3=RobustScaler()
df5=pd.DataFrame(sc3.fit_transform(df1),columns=['id', 'bin_1', 'bin_2', 'City', 'Ord_1','Ord_2','Target'])
df5
```

Encoding.csv :

```
import pandas as pd
df=pd.read_csv("Encoding Data.csv")
df

#feature generation
import category_encoders as ce
be=ce.BinaryEncoder()
ndf=be.fit_transform(df["bin_1"])
df["bin_1"] = be.fit_transform(df["bin_1"])
ndf

ndf2=be.fit_transform(df["bin_2"])
df["bin_2"] = be.fit_transform(df["bin_2"])
ndf2

df1=df.copy()
from sklearn.preprocessing import LabelEncoder,OrdinalEncoder
le=LabelEncoder()
oe=OrdinalEncoder()

df1["nom_0"] = oe.fit_transform(df1[["nom_0"]])
temp=['Cold','Warm','Hot']
oe2=OrdinalEncoder(categories=[temp])
df1['ord_2'] = oe2.fit_transform(df1[['ord_2']])

df1

#feature scaling
from sklearn.preprocessing import MinMaxScaler
sc=MinMaxScaler()
df0=pd.DataFrame(sc.fit_transform(df1),columns=['id', 'bin_1', 'bin_2', 'nom_0','ord_2'])
df0

from sklearn.preprocessing import StandardScaler
sc1=StandardScaler()
df2=pd.DataFrame(sc1.fit_transform(df1),columns=['id', 'bin_1', 'bin_2', 'nom_0','ord_2'])
df2

from sklearn.preprocessing import MaxAbsScaler
sc2=MaxAbsScaler()
df3=pd.DataFrame(sc2.fit_transform(df1),columns=['id', 'bin_1', 'bin_2', 'nom_0','ord_2'])
df3

from sklearn.preprocessing import RobustScaler
sc3=RobustScaler()
df4=pd.DataFrame(sc3.fit_transform(df1),columns=['id', 'bin_1', 'bin_2', 'nom_0','ord_2'])
df4
```

Titanic.csv :

```
import pandas as pd
df=pd.read_csv("titanic_dataset.csv")
df
```

```
#removing unwanted data
df.drop("Name",axis=1,inplace=True)
df.drop("Ticket",axis=1,inplace=True)
df.drop("Cabin",axis=1,inplace=True)

#data cleaning
df.isnull().sum()

df["Age"]=df["Age"].fillna(df["Age"].median())
df["Embarked"]=df["Embarked"].fillna(df["Embarked"].mode()[0])

df.isnull().sum()

df

#feature encoding
from category_encoders import BinaryEncoder
be=BinaryEncoder()
df["Sex"]=be.fit_transform(df[["Sex"]])
ndf=be.fit_transform(df["Sex"])
ndf

df1=df.copy()
from sklearn.preprocessing import LabelEncoder, OrdinalEncoder
embark=['S','C','Q']
e1=OrdinalEncoder(categories=[embark])
df1['Embarked'] = e1.fit_transform(df[['Embarked']])
df1

#feature scaling
from sklearn.preprocessing import MinMaxScaler
sc=MinMaxScaler()
df2=pd.DataFrame(sc.fit_transform(df1),columns=
['Passenger','Survived','Pclass','Sex','Age','SibSp','Parch','Fare','Embarked'])
df2

from sklearn.preprocessing import StandardScaler
sc1=StandardScaler()
df3=pd.DataFrame(sc1.fit_transform(df1),columns=
['Passenger','Survived','Pclass','Sex','Age','SibSp','Parch','Fare','Embarked'])
df3

from sklearn.preprocessing import MaxAbsScaler
sc2=MaxAbsScaler()
df4=pd.DataFrame(sc2.fit_transform(df1),columns=
['Passenger','Survived','Pclass','Sex','Age','SibSp','Parch','Fare','Embarked'])
df4

from sklearn.preprocessing import RobustScaler
sc3=RobustScaler()
df5=pd.DataFrame(sc3.fit_transform(df1),columns=
['Passenger','Survived','Pclass','Sex','Age','SibSp','Parch','Fare','Embarked'])
df5
```

OUTPUT:

Data.csv :

	id	bin_1	bin_2	City	Ord_1	Ord_2	Target
0	0	F	N	Delhi	Hot	High School	0
1	1	F	Y	Bangalore	Warm	Masters	1
2	2	M	N	Mumbai	Very Hot	Diploma	1
3	3	M	Y	Chennai	Cold	Bachelors	0
4	4	M	Y	Delhi	Cold	Bachelors	1
5	5	F	N	Delhi	Very Hot	Masters	0
6	6	M	N	Chennai	Warm	PhD	1
7	7	F	N	Chennai	Hot	High School	1
8	8	M	N	Delhi	Very Hot	High School	0
9	9	F	Y	Delhi	Warm	PhD	0

	bin_1_0	bin_1_1
0	0	1
1	0	1
2	1	0
3	1	0
4	1	0
5	0	1
6	1	0
7	0	1
8	1	0
9	0	1

	bin_2_0	bin_2_1
0	0	1
1	1	0
2	0	1
3	1	0
4	1	0
5	0	1
6	0	1
7	0	1
8	0	1
9	1	0

	id	bin_1	bin_2	City	Ord_1	Ord_2	Target
0	0	0	0	0.0	2.0	0.0	0
1	1	0	1	1.0	1.0	3.0	1
2	2	1	0	0.0	3.0	1.0	1
3	3	1	1	0.0	0.0	2.0	0
4	4	1	1	0.0	0.0	2.0	1
5	5	0	0	0.0	3.0	3.0	0
6	6	1	0	0.0	1.0	4.0	1
7	7	0	0	0.0	2.0	0.0	1
8	8	1	0	0.0	3.0	0.0	0
9	9	0	1	0.0	1.0	4.0	0

	id	bin_1	bin_2	City	Ord_1	Ord_2	Target
0	0.000000	0.0	0.0	0.0	0.666667	0.00	0.0
1	0.111111	0.0	1.0	1.0	0.333333	0.75	1.0
2	0.222222	1.0	0.0	0.0	1.000000	0.25	1.0
3	0.333333	1.0	1.0	0.0	0.000000	0.50	0.0
4	0.444444	1.0	1.0	0.0	0.000000	0.50	1.0
5	0.555556	0.0	0.0	0.0	1.000000	0.75	0.0
6	0.666667	1.0	0.0	0.0	0.333333	1.00	1.0
7	0.777778	0.0	0.0	0.0	0.666667	0.00	1.0
8	0.888889	1.0	0.0	0.0	1.000000	0.00	0.0
9	1.000000	0.0	1.0	0.0	0.333333	1.00	0.0

	id	bin_1	bin_2	City	Ord_1	Ord_2	Target
0	-1.566699	-1.0	-0.816497	-0.333333	0.359211	-1.255555	-1.0
1	-1.218544	-1.0	1.224745	3.000000	-0.538816	0.726900	1.0
2	-0.870388	1.0	-0.816497	-0.333333	1.257237	-0.594737	1.0
3	-0.522233	1.0	1.224745	-0.333333	-1.436842	0.066082	-1.0
4	-0.174078	1.0	1.224745	-0.333333	-1.436842	0.066082	1.0
5	0.174078	-1.0	-0.816497	-0.333333	1.257237	0.726900	-1.0
6	0.522233	1.0	-0.816497	-0.333333	-0.538816	1.387719	1.0
7	0.870388	-1.0	-0.816497	-0.333333	0.359211	-1.255555	1.0
8	1.218544	1.0	-0.816497	-0.333333	1.257237	-1.255555	-1.0
9	1.566699	-1.0	1.224745	-0.333333	-0.538816	1.387719	-1.0

	id	bin_1	bin_2	City	Ord_1	Ord_2	Target
0	0.000000	0.0	0.0	0.0	0.666667	0.00	0.0
1	0.111111	0.0	1.0	1.0	0.333333	0.75	1.0
2	0.222222	1.0	0.0	0.0	1.000000	0.25	1.0
3	0.333333	1.0	1.0	0.0	0.000000	0.50	0.0
4	0.444444	1.0	1.0	0.0	0.000000	0.50	1.0
5	0.555556	0.0	0.0	0.0	1.000000	0.75	0.0
6	0.666667	1.0	0.0	0.0	0.333333	1.00	1.0
7	0.777778	0.0	0.0	0.0	0.666667	0.00	1.0
8	0.888889	1.0	0.0	0.0	1.000000	0.00	0.0
9	1.000000	0.0	1.0	0.0	0.333333	1.00	0.0

	id	bin_1	bin_2	City	Ord_1	Ord_2	Target
0	-1.000000	-0.5	0.0	0.0	0.285714	-0.727273	-0.5
1	-0.777778	-0.5	1.0	1.0	-0.285714	0.363636	0.5
2	-0.555556	0.5	0.0	0.0	0.857143	-0.363636	0.5
3	-0.333333	0.5	1.0	0.0	-0.857143	0.000000	-0.5
4	-0.111111	0.5	1.0	0.0	-0.857143	0.000000	0.5
5	0.111111	-0.5	0.0	0.0	0.857143	0.363636	-0.5
6	0.333333	0.5	0.0	0.0	-0.285714	0.727273	0.5
7	0.555556	-0.5	0.0	0.0	0.285714	-0.727273	0.5
8	0.777778	0.5	0.0	0.0	0.857143	-0.727273	-0.5
9	1.000000	-0.5	1.0	0.0	-0.285714	0.727273	-0.5

	id	bin_1	bin_2	nom_0	ord_2
0	0	F	N	Red	Hot
1	1	F	Y	Blue	Warm
2	2	F	N	Blue	Cold
3	3	F	N	Green	Warm
4	4	T	N	Red	Cold
5	5	T	N	Green	Hot
6	6	F	N	Red	Cold
7	7	T	N	Red	Cold
8	8	F	N	Blue	Warm
9	9	F	Y	Red	Hot

	bin_1_0	bin_1_1
0	0	1
1	0	1
2	0	1
3	0	1
4	1	0
5	1	0
6	0	1
7	1	0
8	0	1
9	0	1

	bin_2_0	bin_2_1
0	0	1
1	1	0
2	0	1
3	0	1
4	0	1
5	0	1
6	0	1
7	0	1
8	0	1
9	1	0

	id	bin_1	bin_2	nom_0	ord_2
0	0	0	0	2.0	2.0
1	1	0	1	0.0	1.0
2	2	0	0	0.0	0.0
3	3	0	0	1.0	1.0
4	4	1	0	2.0	0.0
5	5	1	0	1.0	2.0
6	6	0	0	2.0	0.0
7	7	1	0	2.0	0.0
8	8	0	0	0.0	1.0
9	9	0	1	2.0	2.0

Encoding.csv :

	id	bin_1	bin_2	nom_0	ord_2
0	0.000000	0.0	0.0	1.0	1.0
1	0.111111	0.0	1.0	0.0	0.5
2	0.222222	0.0	0.0	0.0	0.0
3	0.333333	0.0	0.0	0.5	0.5
4	0.444444	1.0	0.0	1.0	0.0
5	0.555556	1.0	0.0	0.5	1.0
6	0.666667	0.0	0.0	1.0	0.0
7	0.777778	1.0	0.0	1.0	0.0
8	0.888889	0.0	0.0	0.0	0.5
9	1.000000	0.0	1.0	1.0	1.0

	id	bin_1	bin_2	nom_0	ord_2
0	-1.566699	-0.654654	-0.5	0.917663	1.324244
1	-1.218544	-0.654654	2.0	-1.376494	0.120386
2	-0.870388	-0.654654	-0.5	-1.376494	-1.083473
3	-0.522233	-0.654654	-0.5	-0.229416	0.120386
4	-0.174078	1.527525	-0.5	0.917663	-1.083473
5	0.174078	1.527525	-0.5	-0.229416	1.324244
6	0.522233	-0.654654	-0.5	0.917663	-1.083473
7	0.870388	1.527525	-0.5	0.917663	-1.083473
8	1.218544	-0.654654	-0.5	-1.376494	0.120386
9	1.566699	-0.654654	2.0	0.917663	1.324244

	id	bin_1	bin_2	nom_0	ord_2
0	0.000000	0.0	0.0	1.0	1.0
1	0.111111	0.0	1.0	0.0	0.5
2	0.222222	0.0	0.0	0.0	0.0
3	0.333333	0.0	0.0	0.5	0.5
4	0.444444	1.0	0.0	1.0	0.0
5	0.555556	1.0	0.0	0.5	1.0
6	0.666667	0.0	0.0	1.0	0.0
7	0.777778	1.0	0.0	1.0	0.0
8	0.888889	0.0	0.0	0.0	0.5
9	1.000000	0.0	1.0	1.0	1.0

	id	bin_1	bin_2	nom_0	ord_2
0	-1.000000	0.000000	0.0	0.285714	0.571429
1	-0.777778	0.000000	1.0	-0.857143	0.000000
2	-0.555556	0.000000	0.0	-0.857143	-0.571429
3	-0.333333	0.000000	0.0	-0.285714	0.000000
4	-0.111111	1.333333	0.0	0.285714	-0.571429
5	0.111111	1.333333	0.0	-0.285714	0.571429
6	0.333333	0.000000	0.0	0.285714	-0.571429
7	0.555556	1.333333	0.0	0.285714	-0.571429
8	0.777778	0.000000	0.0	-0.857143	0.000000
9	1.000000	0.000000	1.0	0.285714	0.571429

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	1	0	3	male	22.0	1	0	7.2500	S
1	2	1	1	female	38.0	1	0	71.2833	C
2	3	1	3	female	26.0	0	0	7.9250	S
3	4	1	1	female	35.0	1	0	53.1000	S
4	5	0	3	male	35.0	0	0	8.0500	S
...	...	...	...	...	...	...	...	...	...
886	887	0	2	male	27.0	0	0	13.0000	S
887	888	1	1	female	19.0	0	0	30.0000	S
888	889	0	3	female	28.0	1	2	23.4500	S
889	890	1	1	male	26.0	0	0	30.0000	C
890	891	0	3	male	32.0	0	0	7.7500	Q

891 rows × 9 columns

	Sex_0	Sex_1
0	0	1
1	1	0
2	1	0
3	1	0
4	0	1
...	...	...
886	0	1
887	1	0
888	1	0
889	0	1
890	0	1

891 rows × 2 columns

```
PassengerId      0
Survived         0
Pclass           0
Sex              0
Age            177
SibSp            0
Parch            0
Fare             0
Embarked         2
dtype: int64
```

Titanic.csv :

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	1	0	3	0	22.0	1	0	7.2500	0.0
1	2	1	1	1	38.0	1	0	71.2833	1.0
2	3	1	3	1	26.0	0	0	7.9250	0.0
3	4	1	1	1	35.0	1	0	53.1000	0.0
4	5	0	3	0	35.0	0	0	8.0500	0.0
...	...	...	...	...	...	...	...	...	...
886	887	0	2	0	27.0	0	0	13.0000	0.0
887	888	1	1	1	19.0	0	0	30.0000	0.0
888	889	0	3	1	28.0	1	2	23.4500	0.0
889	890	1	1	0	26.0	0	0	30.0000	1.0
890	891	0	3	0	32.0	0	0	7.7500	2.0

891 rows × 9 columns

	Passenger	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	0.000000	0.0	1.0	0.0	0.271174	0.125	0.000000	0.014151	0.0
1	0.001124	1.0	0.0	1.0	0.472229	0.125	0.000000	0.139136	0.5
2	0.002247	1.0	1.0	1.0	0.321438	0.000	0.000000	0.015469	0.0
3	0.003371	1.0	0.0	1.0	0.434531	0.125	0.000000	0.103644	0.0
4	0.004494	0.0	1.0	0.0	0.434531	0.000	0.000000	0.015713	0.0
...	...	...	...	...	...	...	...	...	...
886	0.995506	0.0	0.5	0.0	0.334004	0.000	0.000000	0.025374	0.0
887	0.996629	1.0	0.0	1.0	0.233476	0.000	0.000000	0.058556	0.0
888	0.997753	0.0	1.0	1.0	0.346569	0.125	0.333333	0.045771	0.0
889	0.998876	1.0	0.0	0.0	0.321438	0.000	0.000000	0.058556	0.5
890	1.000000	0.0	1.0	0.0	0.396833	0.000	0.000000	0.015127	1.0

891 rows × 9 columns

	Passenger	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	-1.730108	-0.789272	0.827377	-0.737695	-0.565736	0.432793	-0.473674	-0.502445	-0.568837
1	-1.726220	1.266990	-1.566107	1.355574	0.663861	0.432793	-0.473674	0.786845	1.005181
2	-1.722332	1.266990	0.827377	1.355574	-0.258337	-0.474545	-0.473674	-0.488854	-0.568837
3	-1.718444	1.266990	-1.566107	1.355574	0.433312	0.432793	-0.473674	0.420730	-0.568837
4	-1.714556	-0.789272	0.827377	-0.737695	0.433312	-0.474545	-0.473674	-0.486337	-0.568837
...	...	...	...	...	...	...	...	...	...
886	1.714556	-0.789272	-0.369365	-0.737695	-0.181487	-0.474545	-0.473674	-0.386671	-0.568837
887	1.718444	1.266990	-1.566107	1.355574	-0.796286	-0.474545	-0.473674	-0.044381	-0.568837
888	1.722332	-0.789272	0.827377	1.355574	-0.104637	0.432793	2.008933	-0.176263	-0.568837
889	1.726220	1.266990	-1.566107	-0.737695	-0.258337	-0.474545	-0.473674	-0.044381	1.005181
890	1.730108	-0.789272	0.827377	-0.737695	0.202762	-0.474545	-0.473674	-0.492378	2.579199

891 rows × 9 columns

	Passenger	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	0.001122	0.0	1.000000	0.0	0.2750	0.125	0.000000	0.014151	0.0
1	0.002245	1.0	0.333333	1.0	0.4750	0.125	0.000000	0.139136	0.5
2	0.003367	1.0	1.000000	1.0	0.3250	0.000	0.000000	0.015469	0.0
3	0.004489	1.0	0.333333	1.0	0.4375	0.125	0.000000	0.103644	0.0
4	0.005612	0.0	1.000000	0.0	0.4375	0.000	0.000000	0.015713	0.0
...	...	...	...	...	...	...	...	...	...
886	0.995511	0.0	0.666667	0.0	0.3375	0.000	0.000000	0.025374	0.0
887	0.996633	1.0	0.333333	1.0	0.2375	0.000	0.000000	0.058556	0.0
888	0.997755	0.0	1.000000	1.0	0.3500	0.125	0.333333	0.045771	0.0
889	0.998878	1.0	0.333333	0.0	0.3250	0.000	0.000000	0.058556	0.5
890	1.000000	0.0	1.000000	0.0	0.4000	0.000	0.000000	0.015127	1.0

891 rows × 9 columns

	Passenger	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	-1.000000	0.0	0.0	0.0	-0.461538	1.0	0.0	-0.312011	0.0
1	-0.997753	1.0	-2.0	1.0	0.769231	1.0	0.0	2.461242	1.0
2	-0.995506	1.0	0.0	1.0	-0.153846	0.0	0.0	-0.282777	0.0
3	-0.993258	1.0	-2.0	1.0	0.538462	1.0	0.0	1.673732	0.0
4	-0.991011	0.0	0.0	0.0	0.538462	0.0	0.0	-0.277363	0.0
...	...	...	...	...	...	...	...	...	...
886	0.991011	0.0	-1.0	0.0	-0.076923	0.0	0.0	-0.062981	0.0
887	0.993258	1.0	-2.0	1.0	-0.692308	0.0	0.0	0.673281	0.0
888	0.995506	0.0	0.0	1.0	0.000000	1.0	2.0	0.389604	0.0
889	0.997753	1.0	-2.0	0.0	-0.153846	0.0	0.0	0.673281	1.0
890	1.000000	0.0	0.0	0.0	0.307692	0.0	0.0	-0.290356	2.0

891 rows × 9 columns

Result:

Feature Generation process and Feature Scaling process is applied to the given data frames sucessfully.