Esempio Modulo PreProcessing

May 4, 2022

1 Esempi di Utilizzo del MODULO PREPROCESSING della libreria IntelligenzaArtificiale

1.1 Installare la libreria

[ ]:
```
#Per installare la libreria sul tuo computer puoi usare il comando :
pip3 install intelligenzaartificiale

#se invece utilizzi google colab puoi usare:
!pip install intelligenzaartificiale
```

1.2 Importare la libreria

[1]:
```
from intelligenzaartificiale import dataset as dt
from intelligenzaartificiale import preprocessing as pp
```

[2]:
```
# importiamo un Set di dati e stampiamo qualche statistica prima di manipolarlo
il_mio_dataset = dt.leggi_csv("exams.csv")

from intelligenzaartificiale import statistica as st

print("Numero Valori nulli per colonna")
print(st.valori_nan(il_mio_dataset))
print("\n ----------------- \n Percetuali Valori nulli per colonna")
print(st.percentuale_nan(il_mio_dataset))

il_mio_dataset
```

```
Tempo impiegato per leggere il file:  0.0038390159606933594
Numero Valori nulli per colonna
gender                         0
race/ethnicity                 8
parental level of education    7
lunch                          1
test preparation course        5
math score                     3
reading score                  8
writing score                  2
```

```
dtype: int64

 -----------------
 Percetuali Valori nulli per colonna
gender                         0.0
race/ethnicity                 0.8
parental level of education    0.7
lunch                          0.1
test preparation course        0.5
math score                     0.3
reading score                  0.8
writing score                  0.2
dtype: float64
```

[2]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	male	group E	bachelor's degree	standard	none	78.0	59.0	64.0
1	female	group D	some college	free/reduced	none	47.0	52.0	50.0
2	male	group E	high school	free/reduced	none	62.0	47.0	46.0
3	male	group C	master's degree	free/reduced	completed	55.0	65.0	68.0
4	male	NaN	master's degree	free/reduced	none	61.0	54.0	55.0
..	…	…	…	…	…	…	…	…
995	female	group B	high school	standard	none	33.0	36.0	33.0
996	female	group C	some college	standard	none	52.0	59.0	64.0
997	male	group C	some high school	free/reduced	none	66.0	64.0	62.0
998	female	group D	master's degree	standard	completed	99.0	100.0	100.0
999	male	group A	high school	standard	none	46.0	33.0	30.0

[1000 rows x 8 columns]

1.3 Gestire Nulli o Corrotti

[3]:
```
#rimuovere righe con valori nulli o corrotti
il_mio_dataset = pp.rimuovi_nan(il_mio_dataset)

#sostituire valori nulli o corrotti con il valore medio
il_mio_dataset["reading_score_media"] = pp.
 ↪sostituisci_nan_media(il_mio_dataset,"reading score")

#sostituire valori nulli o corrotti con il valore più frequente
il_mio_dataset["reading_score_freq"] = pp.
 ↪sostituisci_nan_frequenti(il_mio_dataset,"reading score")

il_mio_dataset

print("Numero Valori nulli per colonna")
print(st.valori_nan(il_mio_dataset))
print("\n ----------------- \n Percetuali Valori nulli per colonna")
print(st.percentuale_nan(il_mio_dataset))
print("\n ----------------- \n Statistiche di Base")
print(st.statistiche(il_mio_dataset))
```

```
Numero Valori nulli per colonna
gender                         0
race/ethnicity                 0
parental level of education    0
lunch                          0
test preparation course        0
math score                     0
reading score                  0
writing score                  0
reading_score_media            0
reading_score_freq             0
dtype: int64

 -----------------
 Percetuali Valori nulli per colonna
gender                         0.0
race/ethnicity                 0.0
parental level of education    0.0
lunch                          0.0
test preparation course        0.0
math score                     0.0
reading score                  0.0
writing score                  0.0
reading_score_media            0.0
reading_score_freq             0.0
dtype: float64
```

```
 -----------------
 Statistiche di Base
       math score  reading score  writing score  reading_score_media  \
count  966.000000     966.000000     966.000000           966.000000
mean    66.729814      69.210145      68.135611            69.210145
std     15.445122      14.687993      15.267061            14.687993
min      2.000000       3.000000      17.000000             3.000000
25%     57.000000      59.000000      58.000000            59.000000
50%     67.000000      69.000000      68.000000            69.000000
75%     77.750000      80.000000      79.000000            80.000000
max    100.000000     100.000000     100.000000           100.000000

       reading_score_freq
count          966.000000
mean            69.210145
std             14.687993
min              3.000000
25%             59.000000
50%             69.000000
75%             80.000000
max            100.000000
<ipython-input-3-a23882f4e72d>:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  il_mio_dataset["reading_score_media"] =
pp.sostituisci_nan_media(il_mio_dataset,"reading score")
<ipython-input-3-a23882f4e72d>:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  il_mio_dataset["reading_score_freq"] =
pp.sostituisci_nan_frequenti(il_mio_dataset,"reading score")
```

1.4 Gestire gli outliers

[4]:
```
#Rimuovere i valori outlier
il_mio_dataset= pp.rimuovi_outliers(il_mio_dataset,"reading_score_freq")

#Rimuovere i valori outlier e valori nulli
il_mio_dataset = pp.rimuovi_outliers_nan(il_mio_dataset,"reading_score_media")
```

```
il_mio_dataset
print("\n ----------------- \n Statistiche di Base")
print(st.statistiche(il_mio_dataset))
```

```
 -----------------
 Statistiche di Base
       math score  reading score  writing score  reading_score_media  \
count  964.000000     964.000000     964.000000           964.000000
mean    66.791494      69.326763      68.206432            69.326763
std     15.355905      14.470727      15.183609            14.470727
min      2.000000      26.000000      22.000000            26.000000
25%     57.000000      59.000000      58.000000            59.000000
50%     67.000000      69.000000      68.000000            69.000000
75%     78.000000      80.000000      79.000000            80.000000
max    100.000000     100.000000     100.000000           100.000000

       reading_score_freq
count          964.000000
mean            69.326763
std             14.470727
min             26.000000
25%             59.000000
50%             69.000000
75%             80.000000
max            100.000000
```

1.5 Gestire variabili testuali e categoriche

[5]:
```
#effettuare il labelencoding
il_mio_dataset["lunchNew"] = pp.label_encoding(il_mio_dataset,"lunch")

#effettuare il labelencoding su più colonne
il_mio_dataset = pp.label_encoding_multiplo(il_mio_dataset,["test preparation␣
 ↪course", "gender"])

#effettuare il one hot encoding
il_mio_dataset["gender"] = pp.onehot_encoding(il_mio_dataset,"race/ethnicity")

#per rimuovere la vecchia colonna
il_mio_dataset = dt.rimuovi_colonna(il_mio_dataset, "lunch")

il_mio_dataset
```

[5]:
```
     gender race/ethnicity parental level of education  \
0         0        group E           bachelor's degree
1         0        group D                some college
```

```
2         0        group E                 high school
3         0        group C           master's degree
5         0        group D          bachelor's degree
..      …            …                          …
995       0        group B                 high school
996       0        group C               some college
997       0        group C            some high school
998       0        group D           master's degree
999       1        group A                 high school

     test preparation course  math score  reading score  writing score  \
0                          1        78.0           59.0           64.0
1                          1        47.0           52.0           50.0
2                          1        62.0           47.0           46.0
3                          0        55.0           65.0           68.0
5                          1        50.0           58.0           61.0
..                       …          …              …              …
995                        1        33.0           36.0           33.0
996                        1        52.0           59.0           64.0
997                        1        66.0           64.0           62.0
998                        0        99.0          100.0          100.0
999                        1        46.0           33.0           30.0

     reading_score_media  reading_score_freq  lunchNew
0                   59.0                59.0         1
1                   52.0                52.0         0
2                   47.0                47.0         0
3                   65.0                65.0         0
5                   58.0                58.0         1
..                   …                   …       …
995                 36.0                36.0         1
996                 59.0                59.0         1
997                 64.0                64.0         0
998                100.0               100.0         1
999                 33.0                33.0         1

[964 rows x 10 columns]
```

1.6 Normalizzare i dati

[6]:
```
#normalizzare intero datatset
#dataset_normalizzato = pp.normalizza(il_mio_dataset)

#normalizza una specifica colonna
il_mio_dataset["math score"] = pp.normalizza_colonne(il_mio_dataset,"math␣
 ↪score")
```

```
#standardizza intero datatset
#dataset_standardizzato = pp.standardizza(il_mio_dataset)

#standardizza una specifica colonna
il_mio_dataset["writing score"] = pp.
 ↪standardizza_colonne(il_mio_dataset,"writing score")

# dividi i dati in test e train
X_train, X_test, y_train, y_test = pp.dividi_train_test(il_mio_dataset,␣
 ↪"writing score", 0.25 )

print(il_mio_dataset)
print(X_train, X_test, y_train, y_test)
```

```
     gender race/ethnicity parental level of education  \
0         0        group E           bachelor's degree
1         0        group D                some college
2         0        group E                 high school
3         0        group C             master's degree
5         0        group D           bachelor's degree
..      …            …                         …
995       0        group B                 high school
996       0        group C                some college
997       0        group C             some high school
998       0        group D             master's degree
999       1        group A                 high school

     test preparation course  math score  reading score  writing score  \
0                          1         NaN           59.0            NaN
1                          1         NaN           52.0            NaN
2                          1         NaN           47.0            NaN
3                          0         NaN           65.0            NaN
5                          1         NaN           58.0            NaN
..                       …           …              …              …
995                        1         NaN           36.0            NaN
996                        1         NaN           59.0            NaN
997                        1         NaN           64.0            NaN
998                        0         NaN          100.0            NaN
999                        1         NaN           33.0            NaN

     reading_score_media  reading_score_freq  lunchNew
0                   59.0                59.0         1
1                   52.0                52.0         0
2                   47.0                47.0         0
3                   65.0                65.0         0
5                   58.0                58.0         1
..                   …                   …       …
```

```
995                36.0                36.0         1
996                59.0                59.0         1
997                64.0                64.0         0
998               100.0               100.0         1
999                33.0                33.0         1

[964 rows x 10 columns]
     gender race/ethnicity parental level of education  \
636       0        group C            some high school
634       0        group C                some college
496       0        group C                 high school
675       1        group A                some college
11        0        group D            some high school
..      …            …                         …
228       0        group C             master's degree
716       0        group C                 high school
72        0        group C                 high school
638       0        group C            some high school
417       0        group C                some college

     test preparation course  math score  reading score  reading_score_media  \
636                        0         NaN           64.0                 64.0
634                        1         NaN           63.0                 63.0
496                        1         NaN           46.0                 46.0
675                        0         NaN           63.0                 63.0
11                         1         NaN           86.0                 86.0
..                       …           …             …                    …
228                        1         NaN           73.0                 73.0
716                        1         NaN           87.0                 87.0
72                         0         NaN           71.0                 71.0
638                        1         NaN           59.0                 59.0
417                        0         NaN           77.0                 77.0

     reading_score_freq  lunchNew
636                64.0         1
634                63.0         0
496                46.0         0
675                63.0         0
11                 86.0         1
..                  …       …
228                73.0         1
716                87.0         1
72                 71.0         0
638                59.0         0
417                77.0         1

[723 rows x 9 columns]      gender race/ethnicity parental level of education  \
437       0        group D           bachelor's degree
```

```
964       0        group D            some high school
336       0        group C          associate's degree
434       0        group D                some college
910       1        group A                 high school
..      …            …                        …
684       0        group C          associate's degree
843       0        group C           bachelor's degree
574       0        group C            some high school
379       0        group D                some college
558       0        group D           bachelor's degree

     test preparation course  math score  reading score  reading_score_media  \
437                        1         NaN           76.0                 76.0
964                        0         NaN           71.0                 71.0
336                        1         NaN           71.0                 71.0
434                        1         NaN           45.0                 45.0
910                        0         NaN           63.0                 63.0
..                       …           …             …                    …
684                        1         NaN           59.0                 59.0
843                        1         NaN           62.0                 62.0
574                        1         NaN           70.0                 70.0
379                        0         NaN          100.0                100.0
558                        1         NaN           84.0                 84.0

     reading_score_freq  lunchNew
437                76.0         1
964                71.0         1
336                71.0         0
434                45.0         1
910                63.0         1
..                  …       …
684                59.0         1
843                62.0         1
574                70.0         0
379               100.0         1
558                84.0         0

[241 rows x 9 columns] 636   NaN
634   NaN
496   NaN
675   NaN
11    NaN
       ..
228   NaN
716   NaN
72    NaN
638   NaN
417   NaN
```

```
Name: writing score, Length: 723, dtype: float64 437   NaN
964   NaN
336   NaN
434   NaN
910   NaN
       ..
684   NaN
843   NaN
574   NaN
379   NaN
558   NaN
Name: writing score, Length: 241, dtype: float64
```

1.7 Altre risorse

- **Documentazione Ufficiale**

- **Blog Ufficiale**

- **Corsi Gratis**

- **Ebook Gratis**

- **Progetti Python Open Source**

- **Dataset Pubblici**

- **Editor Python Online per il M.L.**

2 Per favore citaci se usi la Libreria.