

Data Preprocessing

The screenshot shows a Google Colab notebook interface. The top bar indicates the notebook is titled 'LR preprocessing.ipynb' and was last edited on November 28. The code editor contains the following Python code:

```
from google.colab import files
uploaded = files.upload()

import io
import pandas as pd

df = pd.read_csv(io.BytesIO(uploaded['Employee_Income.csv']))

df
```

Below the code, a file upload dialog is visible, showing 'Employee_Income.csv' has been uploaded. The output of the code is a preview of the dataset, which is a table with 15 columns and 5 rows of data (plus ellipses indicating more rows).

	age	workclass	fnlwgt	education	education_num	marital_status	occupation	relationship	race	sex	capital_gain	capital_loss	hours_per_week	native_country	income
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40	United-States	<=50K
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13	United-States	<=50K
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40	United-States	<=50K
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	#NAME?	0	0	40	United-States	<=50K
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0	40	Cuba	<=50K

Firstly, to check if the dataset has successfully uploaded or not, I've run the df variable to see the contents of my dataset.

The screenshot shows a Jupyter Notebook interface with two tabs: 'LR preprocessing.ipynb' and 'DV.ipynb'. The active tab is 'LR preprocessing.ipynb'. The code cell contains two commands: `df.head()` and `df.tail()`. The output of `df.head()` displays the first five rows of the dataset, and the output of `df.tail()` displays the last five rows. The dataset has 15 columns: age, workclass, fnlwgt, education, education_num, marital_status, occupation, relationship, race, sex, capital_gain, capital_loss, hours_per_week, native_country, and income. The income column shows values like '<=50K' and '>50K'.

	age	workclass	fnlwgt	education	education_num	marital_status	occupation	relationship	race	sex	capital_gain	capital_loss	hours_per_week	native_country	income
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40	United-States	<=50K
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13	United-States	<=50K
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40	United-States	<=50K
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	#NAME?	0	0	40	United-States	<=50K
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0	40	Cuba	<=50K

	age	workclass	fnlwgt	education	education_num	marital_status	occupation	relationship	race	sex	capital_gain	capital_loss	hours_per_week	native_country	income
4995	43	Private	222971	5th-6th	3	Never-married	Machine-op-inspct	Unmarried	White	Female	0	0	40	Mexico	<=50K
4996	31	Private	259425	HS-grad	9	Married-civ-spouse	Craft-repair	Husband	White	Male	0	0	40	United-States	>50K
4997	47	Self-emp-inc	212120	HS-grad	9	Married-civ-spouse	Craft-repair	Husband	White	Male	0	0	40	United-States	>50K
4998	#NAME?	Private	245880	HS-grad	9	Never-married	Adm-clerical	Not-in-family	White	Male	0	0	60	United-States	<=50K
4999	58	Local-gov	54947	Some-college	10	Never-married	Prof-specialty	Not-in-family	White	Female	0	0	55	United-States	<=50K

To show the first 5 rows of my dataset, I've run the `head()` function. And for showing the last 5 rows, I've used the `tail()` function.

The screenshot shows a Jupyter Notebook interface with a single tab: 'LR preprocessing.ipynb'. The code cell contains the command `df.dtypes`. The output displays the data types for each column of the DataFrame.

Column	dtype
age	object
workclass	object
fnlwgt	object
education	object
education_num	object
marital_status	object
occupation	object
relationship	object
race	object
sex	object
capital_gain	int64
capital_loss	int64
hours_per_week	int64
native_country	object
income	object

By data-type object, 'dtypes', I've checked all the data types of the DataFrame.

```
LR preprocessing.ipynb - Colab
colab.research.google.com/drive/1oV3skb9aE6NinoG5lkbj7h4Cs_UGeBu

LR preprocessing.ipynb
File Edit View Insert Runtime Tools Help Last edited on November 28

+ Code + Text

[] df.describe

<bound method NDFrame.describe of >
0 39 State-gov 77536 Bachelors 13
1 50 Self-emp-not-inc 83311 Bachelors 13
2 38 Private 25546 Hs-grad 9
3 53 Private 234721 11th 7
4 28 Private 58409 Bachelors 13
...
4995 43 Private 222971 5th-6th 3
4996 31 Private 25425 Hs-grad 9
4997 47 Self-emp-inc 112138 Hs-grad 9
4998 4998 Private 245880 Hs-grad 9
4999 58 Local-gov 54947 Some-college 10

marital_status occupation relationship race sex \
0 Never-married Adm-clerical Not-in-family white Male
1 Married-civ-spouse Exec-managerial Husband white Male
2 Divorced Handlers-cleaners Not-in-family white Male
3 Married-civ-spouse Handlers-cleaners Husband black 4998
4 Married-civ-spouse Prof-specialty Wife black Female
...
4995 Never-married Machine-op-inspct Unmarried white Female
4996 Married-civ-spouse Craft-repair Husband white Male
4997 Married-civ-spouse Craft-repair Husband white Male
4998 Never-married Adm-clerical Not-in-family white Male
4999 Never-married Prof-specialty Not-in-family white Female

capital_gain capital_loss hours_per_week native_country income
0 2175 0 40 United-States <50K
1 0 0 13 United-States <50K
2 0 0 40 United-States <50K
3 0 0 40 United-States <50K
4 0 0 40 Cuba <50K
...
4995 0 0 40 United-States <50K
4996 0 0 40 United-States <50K
4997 0 0 40 United-States <50K
4998 0 0 40 United-States <50K
4999 0 0 40 United-States <50K
```

The describe method returns the description of the data in the DataFrame. It contains both numeric & object series of data types.

```
LR preprocessing.ipynb - Colab
colab.research.google.com/drive/1oV3skb9aE6NinoG5lkbj7h4Cs_UGeBu

LR preprocessing.ipynb
File Edit View Insert Runtime Tools Help Last edited on November 28

+ Code + Text

[] df.isnull().sum()

age 0
workclass 0
fnlgt 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
```

The function isnull().sum() checks if the dataset has any null value or not.

```
LR preprocessing.ipynb - Colab
colab.research.google.com/drive/1oV3skb9aE6NinoG5lkbj7h4Cs_UGeBu

LR preprocessing.ipynb
File Edit View Insert Runtime Tools Help Last edited on November 28

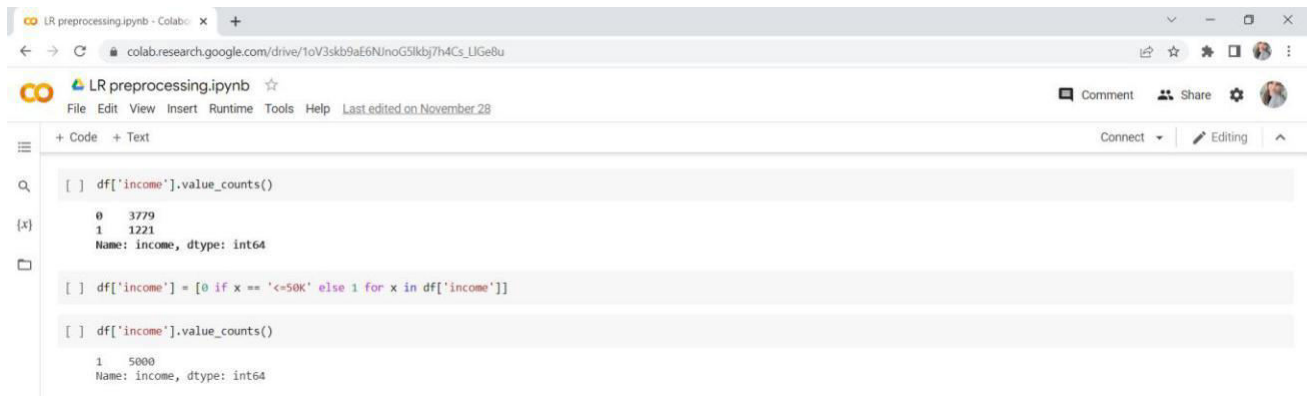
+ Code + Text

[] df.corr()

capital_gain capital_loss hours_per_week
capital_gain 1.000000 -0.033439 0.071881
capital_loss -0.033439 1.000000 0.079426
hours_per_week 0.071881 0.079426 1.000000
```

The corr() function returns the capital_gain and capital_loss.

ENCODING



The screenshot shows a Google Colab notebook interface. The browser address bar displays the URL: `colab.research.google.com/drive/1oV3skb9aE6NinoG5Ikby7h4Cs_UlGe8u`. The notebook title is "LR preprocessing.ipynb". The menu bar includes "File", "Edit", "View", "Insert", "Runtime", "Tools", and "Help", with a note "Last edited on November 28". The interface has tabs for "+ Code" and "+ Text". On the left, there are icons for file explorer, search, and variable inspection. The code cell contains the following Python code:

```
[ ] df['income'].value_counts()

0    3779
1    1221
Name: income, dtype: int64

[ ] df['income'] = [0 if x == '<=50K' else 1 for x in df['income']]

[ ] df['income'].value_counts()

1    5000
Name: income, dtype: int64
```

By applying encoding, we