

## importing necessary packages

```
In [2]: 1 import pandas as pd
        2 import numpy as np
        3 from sklearn.model_selection import train_test_split
        4 from sklearn.linear_model import LinearRegression
        5 import seaborn as sb
```

## importing dataset

```
In [3]: 1 df=pd.read_csv(r"C:\Users\MY HOME\Downloads\car_purchasing (4).csv")
        2 df
```

Out[3]:

	customer name	customer e-mail	country	gender	age
0	Martina Avila	cubilia.Curae.Phasellus@quisaccumsanconvallis.edu	Bulgaria	0	41.8517%
1	Harlan Barnes	eu.dolor@diam.co.uk	Belize	0	40.8706%
2	Naomi Rodriguez	vulputate.mauris.sagittis@ametconsectetueradip...	Algeria	1	43.1528%
3	Jade Cunningham	malesuada@dignissim.com	Cook Islands	1	58.2713%
4	Cedric Leach	felis.ullamcorper.viverra@egetmollislectus.net	Brazil	1	57.3137%
...	...	...	...	...	...
495	Walter	ligula@Cumsociis.ca	Nepal	0	41.4625%
496	Vanna	Cum.sociis.natoque@Sedmolestie.edu	Zimbabwe	1	37.6420%
497	Pearl	penatibus.et@massanonante.com	Philippines	1	53.9434%
498	Nell	Quisque.varius@arcuVivamussit.net	Botswana	1	59.1605%
499	Marla	Camaron.marla@hotmail.com	marlal	1	46.7311%

500 rows × 9 columns



## preprocessing steps

In [4]: 1 df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 9 columns):
#   Column                Non-Null Count  Dtype
---  -
0   customer name         500 non-null    object
1   customer e-mail       500 non-null    object
2   country               500 non-null    object
3   gender               500 non-null    int64
4   age                  500 non-null    float64
5   annual Salary         500 non-null    float64
6   credit card debt      500 non-null    float64
7   net worth             500 non-null    float64
8   car purchase amount   500 non-null    float64
dtypes: float64(5), int64(1), object(3)
memory usage: 35.3+ KB
```

In [5]: 1 df.shape

Out[5]: (500, 9)

In [6]: 1 df.head()

Out[6]:

	customer name	customer e-mail	country	gender	age	
0	Martina Avila	cubilia.Curae.Phasellus@quisaccumsanconvallis.edu	Bulgaria	0	41.851720	6
1	Harlan Barnes	eu.dolor@diam.co.uk	Belize	0	40.870623	6
2	Naomi Rodriquez	vulputate.mauris.sagittis@ametconsectetueradip...	Algeria	1	43.152897	5
3	Jade Cunningham	malesuada@dignissim.com	Cook Islands	1	58.271369	7
4	Cedric Leach	felis.ullamcorper.viverra@egetmollislectus.net	Brazil	1	57.313749	5

In [7]: 1 df.tail()

Out[7]:

	customer name	customer e-mail	country	gender	age	annual Salary
495	Walter	ligula@Cumsociis.ca	Nepal	0	41.462515	71942.40291
496	Vanna	Cum.sociis.natoque@Sedmolestie.edu	Zimbabwe	1	37.642000	56039.49793
497	Pearl	penatibus.et@massanonante.com	Philippines	1	53.943497	68888.77805
498	Nell	Quisque.varius@arcuVivamussit.net	Botswana	1	59.160509	49811.99062
499	Marla	Camaron.marla@hotmail.com	marlal	1	46.731152	61370.67766

```
In [8]: 1 df.isna().any()
```

```
Out[8]: customer name      False
customer e-mail      False
country              False
gender               False
age                  False
annual Salary        False
credit card debt     False
net worth            False
car purchase amount  False
dtype: bool
```

```
In [9]: 1 df.describe()
```

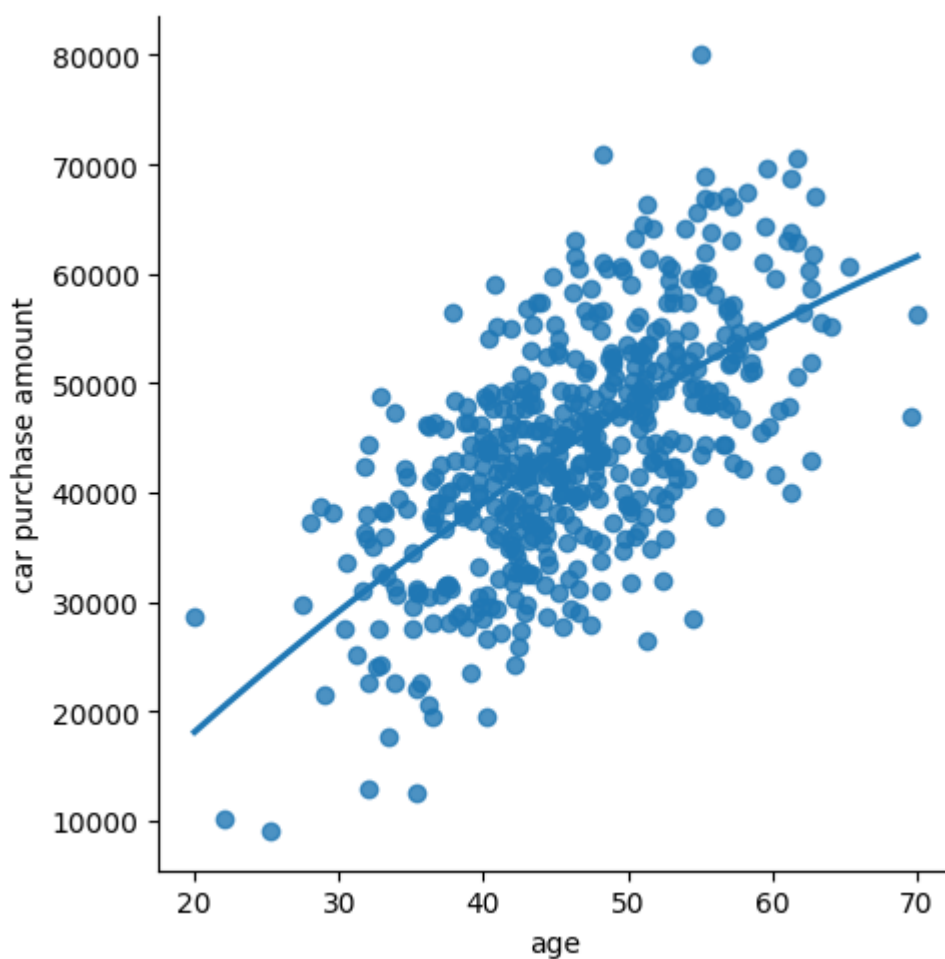
```
Out[9]:
```

	gender	age	annual Salary	credit card debt	net worth	car purchase amount
<b>count</b>	500.000000	500.000000	500.000000	500.000000	500.000000	500.000000
<b>mean</b>	0.506000	46.241674	62127.239608	9607.645049	431475.713625	44209.799218
<b>std</b>	0.500465	7.978862	11703.378228	3489.187973	173536.756340	10773.178744
<b>min</b>	0.000000	20.000000	20000.000000	100.000000	20000.000000	9000.000000
<b>25%</b>	0.000000	40.949969	54391.977195	7397.515792	299824.195900	37629.896040
<b>50%</b>	1.000000	46.049901	62915.497035	9655.035568	426750.120650	43997.783390
<b>75%</b>	1.000000	51.612263	70117.862005	11798.867487	557324.478725	51254.709517
<b>max</b>	1.000000	70.000000	100000.000000	20000.000000	1000000.000000	80000.000000

## visualise the data

```
In [10]: 1 sb.lmplot(x="age",y="car purchase amount",data=df,order=2,ci=None)
```

```
Out[10]: <seaborn.axisgrid.FacetGrid at 0x20f11338d90>
```



## preparing inputs and outputs

```
In [11]: 1 x=df[["gender","age","annual Salary","net worth"]].values  
2 y=df[["car purchase amount"]].values
```

## importing algorithm

```
In [12]: 1 from sklearn.linear_model import LinearRegression
```

## Initialize model

```
In [13]: 1 s=LinearRegression()
```

## Train Model

In [14]: 1 s.fit(x,y)

Out[14]:  
▼ LinearRegression  
LinearRegression()

## accuracy of a model

In [16]: 1 s.score(x,y)

Out[16]: 0.9999999812135176

In [ ]: 1

In [ ]: 1

In [ ]: 1