In [76]:
```python
# import libaries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

In [544]:
```python
x=pd.read_csv(r"C:\Users\user\Downloads\20_states - 20_states.csv")
```

Out[544]:

	id	name	country_id	country_code	country_name	state_code	type	latitude
0	3901	Badakhshan	1	AF	Afghanistan	BDS	NaN	36.734772
1	3871	Badghis	1	AF	Afghanistan	BDG	NaN	35.167134
2	3875	Baghlan	1	AF	Afghanistan	BGL	NaN	36.178903
3	3884	Balkh	1	AF	Afghanistan	BAL	NaN	36.755060
4	3872	Bamyan	1	AF	Afghanistan	BAM	NaN	34.810007
...	...	...	...	...	...	...	...	...
5072	1953	Mashonaland West Province	247	ZW	Zimbabwe	MW	NaN	-17.485103
5073	1960	Masvingo Province	247	ZW	Zimbabwe	MV	NaN	-20.624151
5074	1954	Matabeleland North Province	247	ZW	Zimbabwe	MN	NaN	-18.533157
5075	1952	Matabeleland South Province	247	ZW	Zimbabwe	MS	NaN	-21.052337
5076	1957	Midlands Province	247	ZW	Zimbabwe	MI	NaN	-19.055201

5077 rows × 9 columns

In [545]: `x=x.head(10)`

Out[545]:

	id	name	country_id	country_code	country_name	state_code	type	latitude	lon
0	3901	Badakhshan	1	AF	Afghanistan	BDS	NaN	36.734772	70.8
1	3871	Badghis	1	AF	Afghanistan	BDG	NaN	35.167134	63.7
2	3875	Baghlan	1	AF	Afghanistan	BGL	NaN	36.178903	68.7
3	3884	Balkh	1	AF	Afghanistan	BAL	NaN	36.755060	66.8
4	3872	Bamyan	1	AF	Afghanistan	BAM	NaN	34.810007	67.8
5	3892	Daykundi	1	AF	Afghanistan	DAY	NaN	33.669495	66.0
6	3899	Farah	1	AF	Afghanistan	FRA	NaN	32.495328	62.2
7	3889	Faryab	1	AF	Afghanistan	FYB	NaN	36.079561	64.9
8	3870	Ghazni	1	AF	Afghanistan	GHA	NaN	33.545059	68.4
9	3888	Ghōr	1	AF	Afghanistan	GHO	NaN	34.099578	64.9

In [546]: `x.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   id            10 non-null     int64
 1   name          10 non-null     object
 2   country_id    10 non-null     int64
 3   country_code  10 non-null     object
 4   country_name  10 non-null     object
 5   state_code    10 non-null     object
 6   type          0 non-null      object
 7   latitude      10 non-null     float64
 8   longitude     10 non-null     float64
dtypes: float64(2), int64(2), object(5)
memory usage: 848.0+ bytes
```

In [547]: `x.columns`

Out[547]: `Index(['id', 'name', 'country_id', 'country_code', 'country_name',`
          `'state_code', 'type', 'latitude', 'longitude'],`
          `dtype='object')`

In [556]:
```python
d=x[['id', 'name', 'country_id', 'country_code', 'country_name','latitude']]
```

Out[556]:

	id	name	country_id	country_code	country_name	latitude
0	3901	Badakhshan	1	AF	Afghanistan	36.734772
1	3871	Badghis	1	AF	Afghanistan	35.167134
2	3875	Baghlan	1	AF	Afghanistan	36.178903
3	3884	Balkh	1	AF	Afghanistan	36.755060
4	3872	Bamyan	1	AF	Afghanistan	34.810007
5	3892	Daykundi	1	AF	Afghanistan	33.669495
6	3899	Farah	1	AF	Afghanistan	32.495328
7	3889	Faryab	1	AF	Afghanistan	36.079561
8	3870	Ghazni	1	AF	Afghanistan	33.545059
9	3888	Ghōr	1	AF	Afghanistan	34.099578

In [557]:

Out[557]:

	id	country_id	latitude	longitude
count	10.000000	10.0	10.000000	10.000000
mean	3884.100000	1.0	34.953490	66.458391
std	11.589746	0.0	1.477933	2.579742
min	3870.000000	1.0	32.495328	62.262663
25%	3872.750000	1.0	33.777016	64.905955
50%	3886.000000	1.0	34.988570	66.471945
75%	3891.250000	1.0	36.154067	68.268350
max	3901.000000	1.0	36.755060	70.811995

In [558]:

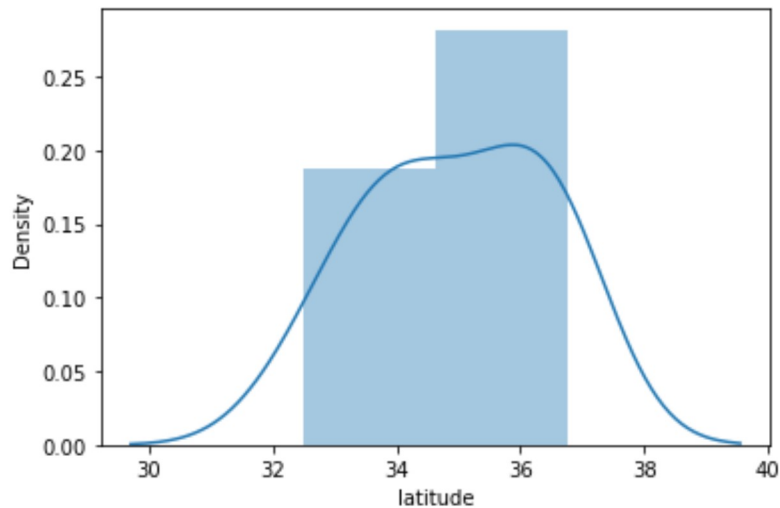Out[558]: <seaborn.axisgrid.PairGrid at 0x190d485f280>
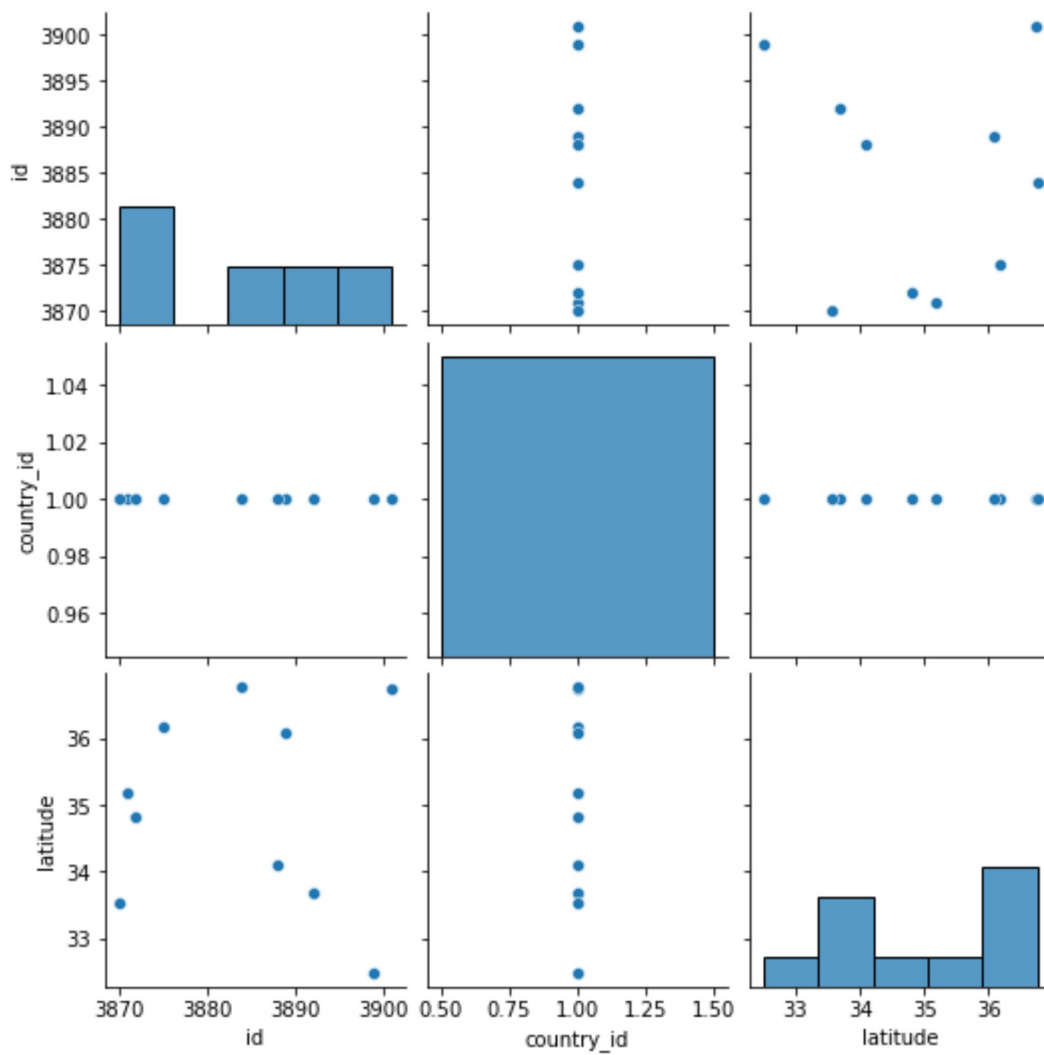
In [566]:

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2557: Fut
ureWarning: `distplot` is a deprecated function and will be removed in a futu
re version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for hi
stograms).
  warnings.warn(msg, FutureWarning)
```
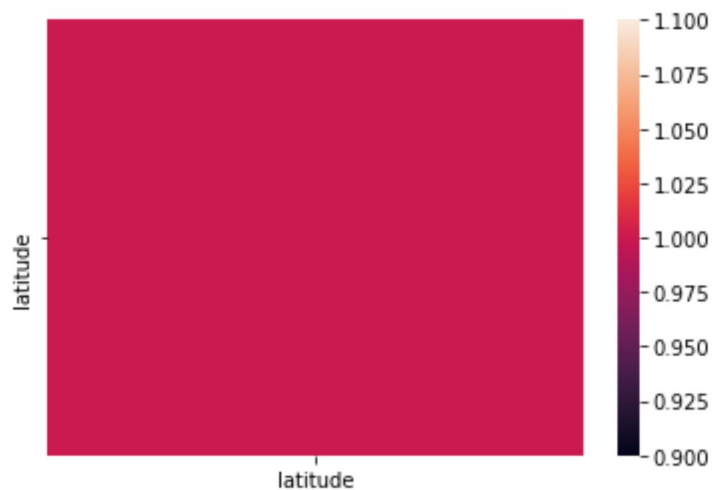
Out[566]: <AxesSubplot:xlabel='latitude', ylabel='Density'>

In [571]:

In [572]:

Out[572]: <AxesSubplot:>

In [573]:
```
x=x1[['latitude']]
```

In [574]:
```python
# to split my dataset into traning and test date

from sklearn.model_selection import train_test_split
```

In [575]:
```python
from sklearn.linear_model import LinearRegression

lr=LinearRegression()
```
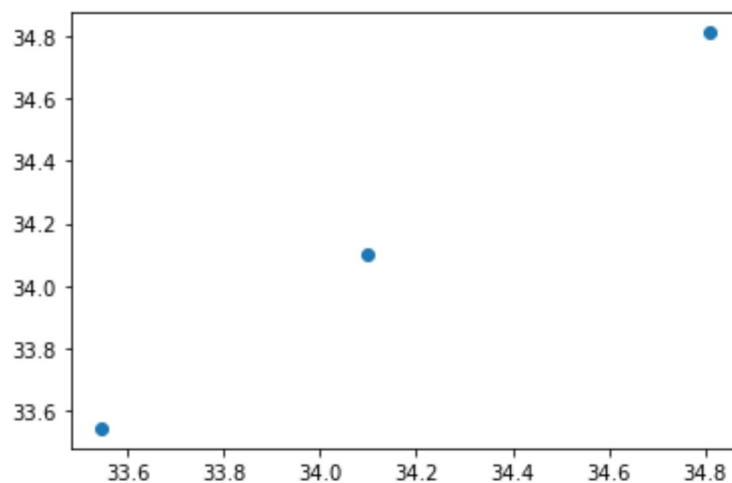
Out[575]: LinearRegression()

In [576]:

0.0

In [577]:
```python
coeff=pd.DataFrame(lr.coef_,x.columns,columns=['Co-efficient'])
```

Out[577]:

	Co-efficient
latitude	1.0

In [578]:
```python
prediction=lr.predict(x_test)
```

Out[578]: <matplotlib.collections.PathCollection at 0x190d50c2e20>

In [579]:

Out[579]: 1.0

In [580]:

Out[580]: 1.0

In [581]:

```
In [582]: rr=Ridge(alpha=10)
          rr.fit(x_train,y_train)
```

Out[582]: 0.1342672818462809

```
In [583]: la=Lasso(alpha=10)
```

Out[583]: Lasso(alpha=10)

```
In [584]:
```

Out[584]: -4.89668396754229

```
In [585]: from sklearn.linear_model import ElasticNet
          en=ElasticNet()
```

Out[585]: ElasticNet()

```
In [586]:
```

Out[586]: array([0.64282594])

```
In [587]:
```

Out[587]: array([34.98401204, 34.52732979, 34.17087066])

```
In [588]:
```

Out[588]: 12.607236922971538

```
In [589]:
```

Out[589]: 0.24774049394817754

```
In [590]:
```

```
In [591]:
```

Mean Absolute Error 0.0

```
In [592]:
```

Mean Squared Error 0.0

```
In [593]:
```

Root Mean Squared Error 0.0

```
In [594]:
```

```
In [595]: filename="prediction"
          pickle.dump(lr,open(filename,'wb'))
```

In [596]:
```python
import pandas as pd
```

In [597]:
```python
filename="prediction"
```

In [600]:
```python
real=[[22],[10]]
```

In [601]:

Out[601]: array([22., 10.])

In [ ]: