In [65]:
```
#Predicting if a person would buy life insurnace based on his age using logistic regression
```

In [66]:
```
import pandas as pd
from matplotlib import pyplot as plt
%matplotlib inline
```

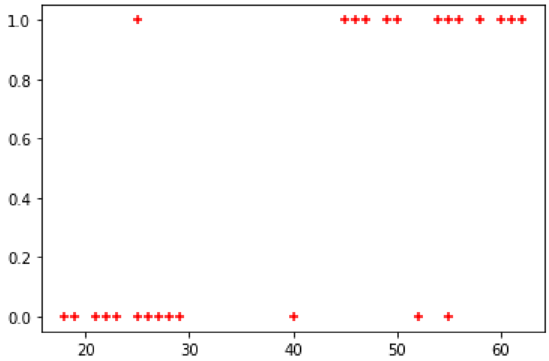
In [5]:
```
df =pd.read_csv("insurance.csv")
df.head()
```

Out[5]:

	age	bought_insurance
0	22	0
1	25	0
2	47	1
3	52	0
4	46	1

In [7]:
```
plt.scatter(df.age,df.bought_insurance,marker ="+",color="red")
```

Out[7]:
```
<matplotlib.collections.PathCollection at 0x1d52dd0da00>
```



In [8]:
```
df.shape
```

Out[8]:
```
(26, 2)
```

In [10]:
```
from sklearn.model_selection import train_test_split
```

In [33]:
```
X_train, X_test, y_train, y_test = train_test_split(df[['age']],df.bought_insurance,train_size=0.8)
```

In [34]:
```
x_test
```

Out[34]:

	age
11	28
25	23
22	45
6	55
13	29
4	46
20	26
2	47
5	56
23	50
7	60
9	61
14	49
8	62
10	18
1	25
18	19
15	55
0	22
16	25
19	21

In [35]:
```
x_train
```

Out[35]:

	age
12	27
3	52
21	40
24	54
17	58

In [36]:
```
from sklearn.linear_model import LogisticRegression
model = LogisticRegression()
```

In [38]:
```
model = LogisticRegression()
```

In [40]:
```
model.fit(X_train, y_train)
```

Out[40]:
```
LogisticRegression()
```

In [25]:
```
model.score(x_test,y_test)
```

Out[25]:
```
1.0
```

In [26]:
```
model.predict_proba(x_test)
```

Out[26]:
```
array([[0.84405234, 0.15594766],
       [0.88638284, 0.11361716],
       [0.78969263, 0.21030737]])
```

In [44]:
```
y_predicted = model.predict(X_test)
```

In [42]:
```
model.predict_proba(X_test)
```

Out[42]:
```
array([[0.86149779, 0.13850221],
       [0.37292323, 0.62707677],
       [0.9470575 , 0.0529425 ],
       [0.42924109, 0.57075891],
       [0.92635703, 0.07364297],
       [0.14054004, 0.85945996]])
```

In [43]:
```
model.score(X_test,y_test)
```

Out[43]:
```
1.0
```

In [45]:
```
y_predicted
```

Out[45]:
```
array([0, 1, 0, 1, 0, 1], dtype=int64)
```

In [46]:
```
X_test
```

Out[46]:

	age
12	27
2	47
10	18
22	45
19	21
17	58

In [ ]:
```
#model.coef_ indicates value of m in y=m*x + b equation
```

In [47]:
```
model.coef_
```

Out[47]:
```
array([[0.11737413]])
```

In [48]:
```
#model.intercept_ indicates value of b in y=m*x + b equation
```

In [49]:
```
model.intercept_
```

Out[49]:
```
array([-4.99688772])
```

In [50]:
```
#Lets defined sigmoid function now and do the math with hand
```

In [52]:
```
import math
def sigmoid(x):
  return 1 / (1 + math.exp(-x))
```

In [57]:
```
def prediction_function(age):
    z = 0.12 * age - 4.99
    y = sigmoid(z)
    return y
```

In [61]:
```
age = 35
prediction_function(age)
```

Out[61]:
```
0.3121686694171596
```

In [59]:
```
#0.312 is less than 0.5 which means person with 35 age will not buy insurance
```

In [62]:
```
age = 43
prediction_function(age)
```

Out[62]:
```
0.542397940774351
```

In [64]:
```
#0.542 is more than 0.5 which means person with 43 will buy the insurance
```

In [ ]:
```

```