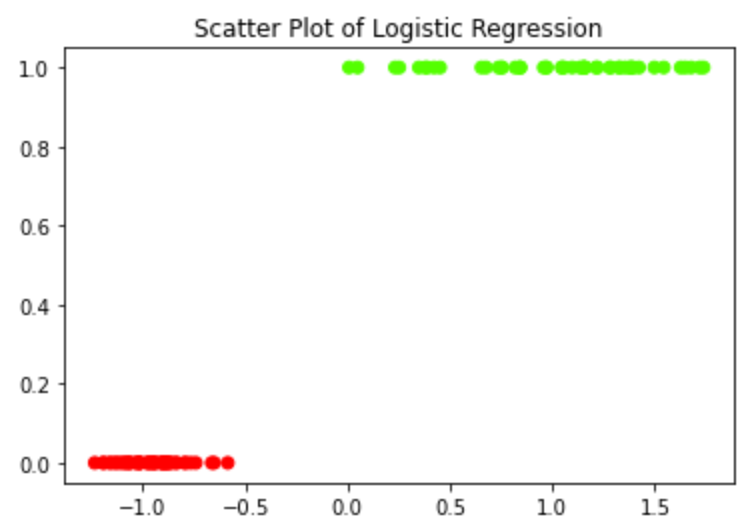
In [26]:
```
from sklearn.datasets import make_classification
from matplotlib import pyplot as plt
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
import pandas as pd
```

In [25]:
```
x, y = make_classification(
    n_samples=100,
    n_features=1,
    n_classes=2,
    n_clusters_per_class=1,
    flip_y=0.03,
    n_informative=1,
    n_redundant=0,
    n_repeated=0
)
```

In [22]:
```
plt.scatter(x, y,c=y, cmap='prism',marker='o')
plt.title('Scatter Plot of Logistic Regression')
plt.show()
```



In [4]:
```
x_train, x_test, y_train, y_test = train_test_split(x, y, random_state=1)
```

In [5]:
```
log_reg = LogisticRegression()
log_reg.fit(x_train, y_train)
```

Out[5]:
```
LogisticRegression()
```

In [7]:
```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
          intercept_scaling=1, max_iter=100, multi_class='ovr', n_jobs=1,
          penalty='l2', random_state=None, solver='liblinear', tol=0.0001,
          verbose=0, warm_start=False)
```

Out[7]:
```
LogisticRegression(multi_class='ovr', n_jobs=1, solver='liblinear')
```

In [9]:
```
print(log_reg.coef_)
print(log_reg.intercept_)
```

```
[[3.21633667]]
[0.13958327]
```

In [11]:
```
y_pred = log_reg.predict(x_test)
```

In [12]:
```
confusion_matrix(y_test, y_pred)
```

Out[12]:
```
array([[10,  0],
       [ 0, 15]], dtype=int64)
```

In [ ]:
```

```