

```
import numpy as np
from matplotlib import pyplot as plt
import scipy.stats as stats
```

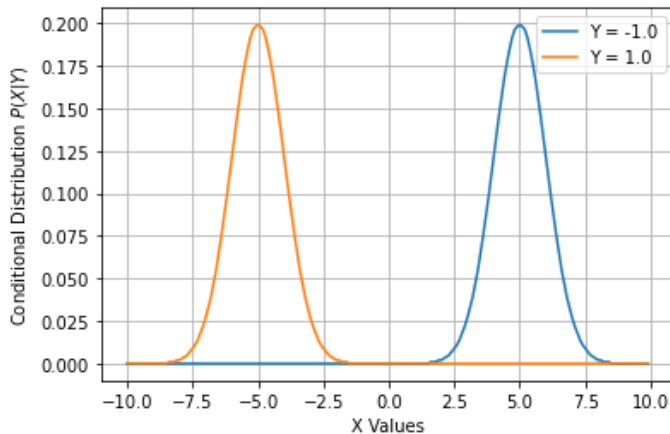
```
def evaluate_conditional_distribution(x, y):
    return (1.0 / (2 * np.sqrt(2 * np.pi))) * np.exp(-(x + 5 * y) ** 2 / 2.0))
```

```
vectorized_conditional = np.vectorize(evaluate_conditional_distribution)
```

```
X_array = np.arange(-10.0, 10.0, 0.1)
Y_array = [-1.0, 1.0]
```

```
for y in Y_array:
    evaluated_distribution = vectorized_conditional(X_array, y).round(3)
    plt.plot(X_array, evaluated_distribution, label="Y = " + str(y))
```

```
plt.grid()
plt.legend()
plt.xlabel("X Values")
plt.ylabel(r"Conditional Distribution $P(X | Y)$")
plt.show()
```



```
print("Classification Error of Bayes Optimal Classifier described in (c): {:.3e}".format(
    1 - stats.norm.cdf(5, loc=0, scale=1)))
```

```
Classification Error of Bayes Optimal Classifier described in (c): 2.867e-07
```