Praticle 4 : Implementation of Logistic Regression

In [67]:
```
%matplotlib inline
import io
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import sklearn.linear_model
```

In [68]:
```
# Load Data
data =  pd.read_csv('grain.csv')
```

In [69]:
```
# Train logistic regression model
X = df[['Grain size (mm)']]
y = df['Spiders']
clf = sklearn.linear_model.LogisticRegression()
clf.fit(X, y)
```

Out[69]:
```
▾ LogisticRegression

LogisticRegression()
```

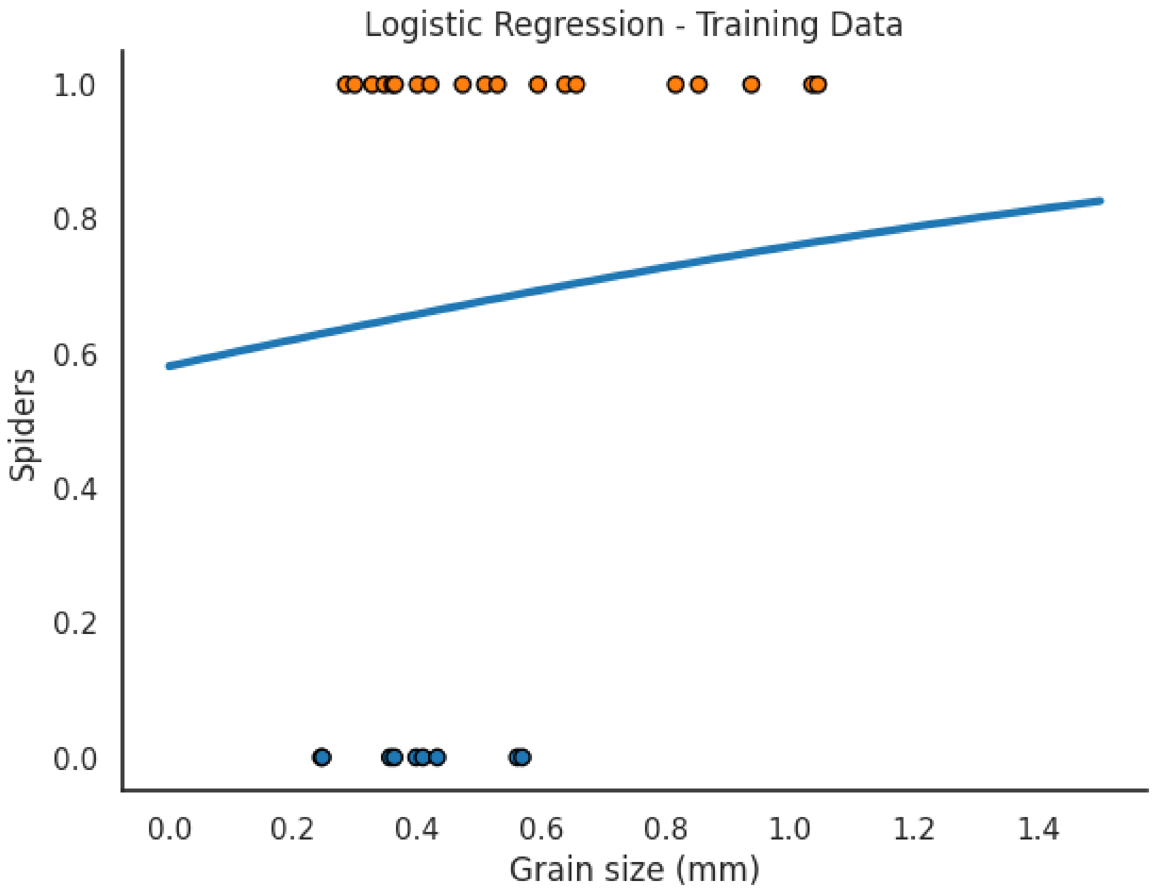
In [70]:
```
def plot_log_reg(x, y, data, clf, xmin=None, xmax=None, alpha=1, ax=None)
    if ax is None:
        fig, ax = plt.subplots()
    else:
        fig = ax.figure
    ax.scatter(data[x], data[y], c=data[y], cmap=ListedColormap(['tab:blu
    if xmin is None:
        xmin = x.min()
    if xmax is None:
        xmax = x.max()
    X_test = np.linspace(xmin, xmax, 300)

    loss = scipy.special.expit(X_test * clf.coef_ + clf.intercept_).ravel
    ax.plot(X_test, loss, linewidth=3)

    ax.set_xlabel(x)
    ax.set_ylabel(y)
    fig.tight_layout()
    sns.despine()
    return fig, ax
```

In [71]:
```
# Plot logistic regression line for training data
fig, ax = plot_log_reg(x='Grain size (mm)', y='Spiders', data=df, clf=clf
ax.set_title("Logistic Regression - Training Data")
```

Out[71]:
```
Text(0.5, 1.0, 'Logistic Regression - Training Data')
```

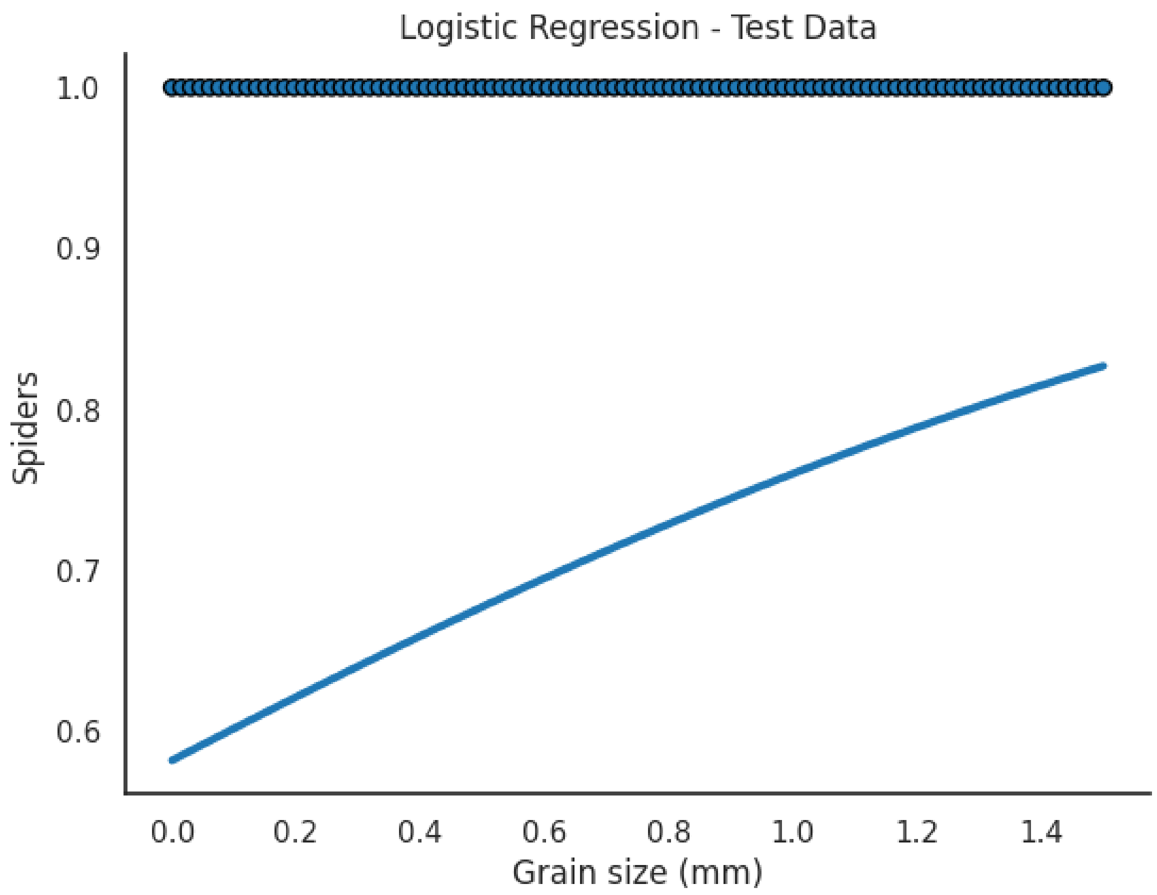


In [72]:
```
# Generate some test data
X_test = np.linspace(0, 1.5, 100).reshape(-1, 1)
y_test_pred = clf.predict(X_test)
test_data = pd.DataFrame({'Grain size (mm)': X_test.flatten(), 'Spiders':
```

```
/home/geek/.pyenv/versions/3.9.10/lib/python3.9/site-packages/sklearn/bas
e.py:439: UserWarning: X does not have valid feature names, but LogisticRe
gression was fitted with feature names
  warnings.warn(
```

In [73]:
```
# Plot logistic regression line for test data
fig, ax = plot_log_reg(x='Grain size (mm)', y='Spiders', data=test_data,
ax.set_title("Logistic Regression - Test Data")
```

Out[73]:
```
Text(0.5, 1.0, 'Logistic Regression - Test Data')
```



```
In [74]: plt.show()
```

Steps followed while Implementation:

- The code reads the dataset from a CSV file named 'grain.csv' using pandas.
- It trains a logistic regression model on the 'Grain size (mm)' and 'Spiders' columns of the dataset.
- The function plot_log_reg is defined to visualize the logistic regression line on the scatter plot of the data.
- The logistic regression line is plotted for the training data, and a separate plot is generated for the test data.
- The resulting plots show the data points as black dots and the logistic regression line as a curve.