1. Logistic Regression Model:

Logistic regression is a statistical technique used to model the relationship between a binary or categorical dependent variable and one or more independent variables. Unlike linear regression, which is used to predict continuous outcomes, logistic regression is used to predict binary or categorical outcomes, such as whether an event will occur or not (e.g., whether a customer will purchase a product or not).

The logistic regression model can be represented mathematically as:

$$P(Y=1|X) = 1 / (1 + e^{(-\beta_0 - \beta_1X_1 - \beta_2X_2 - \ldots - \beta_nX_n)})$$

Where:

- $P(Y=1|X)$ is the probability that the dependent variable Y is equal to 1, given the values of the independent variables X
- e is the base of the natural logarithm (approximately 2.718)
- β_0 is the intercept (the value of the logit when all Xs are 0)
- $\beta_1, \beta_2, \ldots, \beta_n$ are the regression coefficients, which represent the change in the logit (the natural logarithm of the odds ratio) for a one-unit change in the corresponding X, while holding all other Xs constant

The logistic regression model is based on the logit function, which is the natural logarithm of the odds ratio. The odds ratio is the ratio of the probability of the event occurring (Y=1) to the probability of the event not occurring (Y=0).

Logistic regression can be used for various purposes, such as:

- Predicting the probability of a binary or categorical outcome based on the values of the independent variables
- Identifying the relative importance of the independent variables in explaining the variation in the dependent variable
- Evaluating the strength and direction of the relationship between the dependent and independent variables

Logistic regression can be extended to handle multiple independent variables (multiple logistic regression) and can also be used with non-linear relationships by transforming the variables (e.g., using polynomial or logarithmic functions).

2. Creating a Dataset:

Here's an example dataset of 30 records with features "road length", "ticket price", and "number of passengers":

```
import pandas as pd
import numpy as np

# Generate random data
road_length = np.random.uniform(10, 100, 30)
ticket_price = np.random.uniform(20, 50, 30)
num_passengers = np.random.randint(10, 100, 30)

# Create the dataset
data = {'road_length': road_length, 'ticket_price': ticket_price, 'num_passengers': num_passengers}
df = pd.DataFrame(data)

print(df.head())
```

3. Practical Example of Logistic Regression:

Let's assume we want to use logistic regression to predict whether a customer will purchase a product or not based on the road length and ticket price. Here's the Python code:

```
import pandas as pd
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report


# Load the dataset
df = pd.DataFrame({
    'road_length': [10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100],
    'ticket_price': [25, 28, 31, 34, 37, 40, 43, 46, 49, 52, 25, 28, 31, 34, 37, 40, 43, 46, 49],
    'purchased': [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1]
})


# Split the dataset into training and testing sets
X = df[['road_length', 'ticket_price']]
y = df['purchased']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


# Create and train the logistic regression model
model = LogisticRegression()
model.fit(X_train, y_train)


# Make predictions on the test set
y_pred = model.predict(X_test)


# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
cm = confusion_matrix(y_test, y_pred)
report = classification_report(y_test, y_pred)


print("Accuracy:", accuracy)
print("Confusion Matrix:\n", cm)
print("Classification Report:\n", report)
print("Regression Coefficients:", model.coef_)

print("Intercept:", model.intercept_)
```

In this example, we first load the dataset, which contains the "road length", "ticket price", and "purchased" (a binary variable indicating whether the customer purchased the product or not) features. We then split the dataset into training and testing sets using the `train_test_split` function from scikit-learn.

Next, I created a `LogisticRegression` model, fit it to the training data, and use it to make predictions on the test data. Finally, we evaluate the model's performance by calculating the accuracy score, confusion matrix, and classification report.

The regression coefficients and intercept are also printed, which can be used to interpret the model's results. For instance, the regression coefficient for "road length" represents the change in the log-odds of the customer purchasing the product for a one-unit increase in road length, while holding the ticket price constant.

The logistic regression model can be used to predict the probability of a binary outcome, such as whether a customer will purchase a product or not, based on the values of the independent variables. The predicted probabilities can be used to make decisions, such as whether to offer a discount or not, based on the likelihood of a customer making a purchase.