```
import numpy as np
import pandas as pd
```

```
df = pd.read_csv('Parole-2.csv')
```

a) How many parolees do we have data for? Of the parolees that we have data for, what percentage violated the terms of their parole?

```
total_records = len(df)
print("Total number of records:")
print(total_records)
violators = len(df[df['Violator'] == 1])/total_records*100
print("Percentage of violators:")
print(violators)
```

```
Total number of records:
675
Percentage of violators:
11.555555555555555
```

Number of paralees = 675

% of violators = 11.55

b) Randomly split the data into a training set and a testing set, putting 70% of the data in the training set. Then, build a logistic regression model to predict the variable Violator using all of the other variables as independent variables. You should use the training dataset to build the model.

```
from sklearn.model_selection import train_test_split
df_train, df_test = train_test_split(df, test_size=0.3, random_state=42, stratify=df['Violator'])
```

```
import statsmodels.formula.api as smf
categorical_cols = df_train.select_dtypes(include='object').columns
print(categorical_cols)
X_column_names = [col for col in df_train.columns if col != 'Violator']
formula_terms = []
for col in X_column_names:
    if col in categorical_cols:
        formula_terms.append(f"C({col})")
    else:
        formula_terms.append(col)

formula_str = "Violator ~ " + " + ".join(formula_terms)
logit_reg = smf.logit(formula=formula_str, data=df_train)
logit_reg_results = logit_reg.fit()
print(logit_reg_results.summary())
```

```
Index(['State', 'Crime'], dtype='object')
Optimization terminated successfully.
         Current function value: 0.248365
         Iterations 8
                           Logit Regression Results
==============================================================================
Dep. Variable:               Violator   No. Observations:                  472
Model:                          Logit   Df Residuals:                      459
Method:                           MLE   Df Model:                           12
Date:                Sun, 09 Nov 2025   Pseudo R-squ.:                  0.3100
Time:                        03:27:43   Log-Likelihood:                -117.23
converged:                       True   LL-Null:                       -169.89
Covariance Type:            nonrobust   LLR p-value:                 4.995e-17
================================================================================================
                               coef    std err          z      P>|z|      [0.025      0.975]
```

```
-------------------------------------------------------------------------------------
Intercept               -2.9017      1.466     -1.979      0.048      -5.775      -0.028
C(State)[T.Louisiana]    0.7345      0.582      1.262      0.207      -0.406       1.875
C(State)[T.Other]       -0.0287      0.498     -0.058      0.954      -1.004       0.947
C(State)[T.Virginia]    -3.5855      0.714     -5.024      0.000      -4.984      -2.187
C(Crime)[T.Drugs]        0.5706      0.735      0.777      0.437      -0.869       2.011
C(Crime)[T.Larceny]      1.2819      0.776      1.652      0.098      -0.239       2.802
C(Crime)[T.Other]        0.7824      0.694      1.127      0.260      -0.578       2.143
Male                    -0.1182      0.427     -0.277      0.782      -0.955       0.718
RaceWhite               -1.1683      0.395     -2.957      0.003      -1.943      -0.394
Age                      0.0161      0.017      0.954      0.340      -0.017       0.049
TimeServed              -0.1341      0.125     -1.071      0.284      -0.379       0.111
MaxSentence              0.0760      0.055      1.390      0.165      -0.031       0.183
MultipleOffenses         1.4090      0.394      3.574      0.000       0.636       2.182
=====================================================================================
```

i) Describe your resulting model. Which variables are significant in your model?

Ans: Significant variables while predicting the violators are:

1. State (when it is Virginia) : Significant negative coefficient indicates that, it is less likely that parole in Virginia can be violator
2. RaceWhite : negative coefficient shows that white parolees are less likely to violate parole than non-white parolees
3. Multiple offences : Positive coefficient indicates paroles with record of multiple offeces can be violators

ii) Consider a parolee who is male, of white race, aged 50 years at prison release, from the state of Maryland, served 3 months, had a maximum sentence of 12 months, did not commit multiple offenses, and committed a larceny. According to your model, what is the probability that this individual is a violator? (HINT: You should use the coefficients of your model and the Logistic Response Function to solve this problem.)

```
new_parolee_data = pd.DataFrame({
    'Male': [1],
    'RaceWhite': [1],
    'Age': [50],
    'State': ['Other'],
    'TimeServed': [3],
    'MaxSentence': [12],
    'MultipleOffenses': [0],
    'Crime': ['Larceny']
})

probability = logit_reg_results.predict(new_parolee_data)[0]

print(f"The predicted probability that this individual is a violator is: {probability:.4f}")
```

```
The predicted probability that this individual is a violator is: 0.1652
```

ANS: probability that this individual is a violator = 0.1652

iii) Now compute the model's predicted probabilities for parolees in the testing set. Then create a confusion matrix for the test set using a threshold of 0.5. What is the model's false positive rate on the test set? False negative rate? Overall accuracy?

```
df_test['PredViolatorProb'] = logit_reg_results.predict(df_test)
df_test['PredViolator'] = (df_test['PredViolatorProb'] > 0.5)*1
pd.crosstab( df_test['Violator'],df_test['PredViolator'])
```

PredViolator	0	1
Violator		
0	174	6
1	16	7

ANS:

False positive rate = (6)/(6+174) = 0.033

False negative rate = (16)/(16+7) = 0.6957

Accuracy = (174+7)/(16+6+174+7) = 0.8916

iv) Compare your accuracy on the test set to a baseline model that predicts every parolee in the test set is a non-violator, regardless of the values of the independent variables. Does your model improve over this simple model?

```
pd.crosstab(df_test['Violator'],0)
```

col_0	0
Violator	
0	180
1	23

ANS:

Accuracy of baseline = 180/(180+23) = 0.8867

Accuracy of logistic model = 0.8916

Accuracy of logistic model is improved over baseline model.

v) Consider a parole board that might use your model to predict whether parolees will be violators or not. The job of a parole board is to make sure that a prisoner is ready to be released into free society, and therefore parole boards tend to be particularly concerned about releasing prisoners who will violate their parole. Would the parole board be more concerned by false positive errors or false negative errors? How should they adjust their threshold to reflect their error preferences?

ANS

Parole board should be concerned about "False Negative" errors because model predicts prisonor is non-violator but actually can be violator. By lowering predictionthreashold can lower the errors in False Negative rate

```
df_test['PredViolatorProb'] = logit_reg_results.predict(df_test)
df_test['PredViolator'] = (df_test['PredViolatorProb'] > 0.3)*1
pd.crosstab( df_test['Violator'],df_test['PredViolator'])
```

PredViolator	0	1
Violator		
0	165	15
1	14	9

vi) Compute the AUC of the model on the test set, and interpret what the number means in this context. Considering the AUC, the accuracy compared to the baseline model, and what happens when the threshold is adjusted, do you think this model is of value to a parole board? Why or why not?

```
from sklearn.metrics import roc_auc_score

# Compute AUC
auc_score = roc_auc_score(df_test['Violator'], df_test['PredViolatorProb'])

print(f"Area Under the Curve (AUC) on the test set: {auc_score:.4f}")
```

```
Area Under the Curve (AUC) on the test set: 0.8024
```

ANS:

- The Area Under the Curve (AUC) for the model on the test set was 0.8024, indicating a reasonably good ability of model to distinguish between violators and non-violators.
- Considering AUC(0.8024), accuracy compared to baseline model(0.8916 over 0.8867) and decrease in False Negative rate after lowering the threashold indicates that this model does add value if threshold kept less considering the concern of the board about releasing violators.

c) Our goal in this problem has been to predict the outcome of a parole decision, and we used a publicly available dataset of parole releases for predictions. It is always important to evaluate a dataset for possible sources of bias, especially when the dataset only contains a subset of the observations of interest. The Parole.csv dataset contains all individuals released from parol However, it does not contain parolees who neither violated their parole nor completed their term in 2004, causingnon-violators to be underrepresented. This is called selection bias or selecting on the dependent variable, because we used our dependent variable (parole violation) to select only a subset of all relevant parolees to include in our analysis. How could we improve our dataset to best address selection bias?

ANS

When non-violators are underrepresented because of how the data was collected (only including those who either violated or completed their term, and missing those who neither), model could not learn the true relationship between the independent variables and the likelihood of violation.

To best address this selection bias, we would need to collect data on all individuals released on parole, regardless of their outcome in 2004. Specifically, the improved dataset should include:

1. All Parolees Released: This means including individuals who were released on parole and were still on parole at the end of 2004, but had not yet violated or completed their term. These individuals represent the observations that are missing from the current dataset.
2. Their Full Outcomes: For all parolees, we would need to track their full parole outcome (violation, successful completion, or still on parole) over a consistent observation period, after the initial release decision.

This would allow the model to learn from a complete and unbiased sample of parole outcomes. In essence, the goal is to shift from a dataset that only includes observed outcomes to one that includes all exposures to parole and their subsequent outcomes. This would provide a more representative sample of both violators and non-violators, allowing the model to more accurately estimate the probability of violation for any given parolee.