```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import statsmodels.api as sm
        import statistics
        from sklearn.linear_model import LinearRegression
        from sklearn.linear_model import LogisticRegression
        from sklearn.discriminant_analysis import QuadraticDiscriminantAnalysis
        from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
        from sklearn.neighbors import KNeighborsClassifier
        from sklearn.metrics import confusion_matrix
        from sklearn.model_selection import train_test_split
        from sklearn.model_selection import LeaveOneOut
```

```
In [2]: df = pd.read_csv('Boston.csv')
        df_copy = df.copy()
        df_copy = df_copy.drop('Unnamed: 0', 1)
        # crim = per capita crime rate by town
        # zn = proportion of residential land zoned for lots over 25,000 sq.ft.
        # INDUS - proportion of non-retail business acres per town.
        # CHAS - Charles River dummy variable (1 if tract bounds river; 0 otherwise)
        # NOX - nitric oxides concentration (parts per 10 million)
        # RM - average number of rooms per dwelling
        # AGE - proportion of owner-occupied units built prior to 1940
        # DIS - weighted distances to five Boston employment centres
        # RAD - index of accessibility to radial highways
        # TAX - full-value property-tax rate per $10,000
        # PTRATIO - pupil-teacher ratio by town
        # B - 1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town
        # LSTAT - % lower status of the population
        # MEDV - Median value of owner-occupied homes in $1000's
```

Problem a

```
In [5]: # Estimated Population Mean of 'medv'
        mu_hat = df_copy['medv'].mean()
        print('The estimated population mean of "medv" is ' + repr(mu_hat))
```

```
The estimated population mean of "medv" is 22.532806324110698
```

Problem b

In [10]:
```python
num_observations = df_copy['medv'].shape[0]
std_mu_hat = df_copy['medv'].std()
std_err_mu_hat = std_mu_hat/np.sqrt(num_observations)
print('The estimated standard error of the population mean of "medv" is ' + re
pr(std_err_mu_hat))
```

The estimated standard error of the population mean of "medv" is 0.4088611474
975351

Problem c

In [27]:
```python
from sklearn.utils import resample

# BOOTSTRAPPING
medv_data = np.asarray(df_copy['medv'])
num_boot_samples = 20
sample_size = df_copy.shape[0]
sample_means_list = []

for i in range(num_boot_samples):
    bootstrap_sample = resample(medv_data)
    current_sample_mean = np.average(bootstrap_sample)
    sample_means_list.append(current_sample_mean)

std_err_mu = np.std(sample_means_list)
std_err_mu
```

Out[27]: 0.38941128639910266

Problem d

In [28]:
```python
print('[' + repr(mu_hat - 2*std_err_mu) + ', ' + repr(mu_hat + 2*std_err_mu) +
']')
```

[21.753983751312493, 23.311628896908903]

In [ ]: