**Assignment 1** (15%)

CSE 5120 – Introduction to Artificial Intelligence – Fall 2024

*Submitted to*

Department of Computer Science and Engineering  
California State University, San Bernardino, California

*by*

Cameron Hartling (007735059)

Date: November 15, 2024

*Email:*

* 007735059@coyote.csusb.edu

**Assignment Report**

The disadvantaged communities dataset lists California cities and various data that might contribute to how disadvantaged the cities are. The data is labeled with the DAC Category which I will be using as the target variable. All of the variables afterwards will be used as the attributes.

dac = pd.read\_csv('disadvantaged\_communities.csv')

I loaded in the data using the pandas read\_csv() function.

dac.dropna(inplace = True)

I cleaned the data by removing all of the rows that had missing values.

from sklearn.preprocessing import OrdinalEncoder

enc = OrdinalEncoder()

dac[["CES 4.0 Percentile Range","DAC category"]] = enc.fit\_transform(dac[["CES 4.0 Percentile Range","DAC category"]])

To encode the categorical variables into ordinal integers I used the OrdinalEncoder from the sklearn library.

X = dac.iloc[:, 11:]

y = dac.iloc[:, 9]

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=0, stratify = y)

I split the variables into a training set of 80% of the rows and a test set of 20% of the rows.

from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

X\_train = sc.fit\_transform(X\_train)

X\_test = sc.fit\_transform(X\_test)

I standardized the data by using the standard scalar.

from sklearn.svm import SVC

svc\_model = SVC(kernel = 'rbf', random\_state = 0, C = 10.0, gamma = 0.3)

svc\_model.fit(X\_train, y\_train)

y\_pred = svc\_model.predict(X\_test)

The SVM classifier is made using the radial basis function.

The accuracy scores of the two methods were SVM at 0.3507 and Random Forest at 0.6537. This shows that the Random Forest model was a better fit than the SVM model.