P2 – SUPERVISED LEARNING

K-NN

Characteristics of a KNN Model

- Fast to create model because it simply stores data
- Slow to predict because many distance calculations
- Can require lots of memory if data set is large

K Nearest Neighbors: The Syntax

Import the class containing the classification method

```
from sklearn.neighbors import KNeighborsClassifier
```

K Nearest Neighbors: The Syntax

Import the class containing the classification method

```
from sklearn.neighbors import KNeighborsClassifier
```

Create an instance of the class

```
KNN = KNeighborsClassifier(n_neighbors=3)
```

K Nearest Neighbors: The Syntax

Import the class containing the classification method

```
from sklearn.neighbors import KNeighborsClassifier
```

Create an instance of the class

```
KNN = KNeighborsClassifier(n_neighbors=3)
```

Fit the instance on the data and then predict the expected value

```
KNN = KNN.fit(X_data, y_data)

y_predict = KNN.predict(X_data)
```

K Nearest Neighbors: The Syntax

Import the class containing the classification method

```
from sklearn.neighbors import KNeighborsClassifier
```

Create an instance of the class

```
KNN = KNeighborsClassifier(n_neighbors=3)
```

Fit the instance on the data and then predict the expected value

```
KNN = KNN.fit(X_data, y_data)
y_predict = KNN.predict(X_data)
```

The fit and predict/transform syntax will show up throughout the course.

K Nearest Neighbors: The Syntax

Import the class containing the classification method

```
from sklearn.neighbors import KNeighborsClassifier
```

Create an instance of the class

```
KNN = KNeighborsClassifier(n_neighbors=3)
```

Fit the instance on the data and then predict the expected value

```
KNN = KNN.fit(X_data, y_data)
y_predict = KNN.predict(X_data)
```

Regression can be done with KNeighborsRegressor.