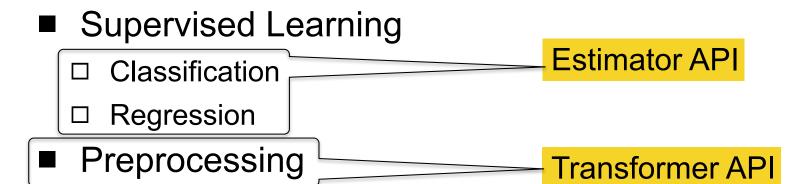
# Bluffers guide to scikit-learn

## **Sarah Jane Delany**

Slides based on material from Padraig Cunningham





### Classification in scikit-learn

```
forecast = pd.read_csv('Forecast.csv')
y = forecast.pop('Go-Out').values
X = forecast.values
```

|   | Temperature  |    | Humidity | Wind_Speed | Go-Out |   |
|---|--------------|----|----------|------------|--------|---|
| 0 |              | 6  | 85       | 30         |        | 0 |
| 1 |              | 14 | 90       | 35         |        | 0 |
| 2 |              | 15 | 86       | 8          |        | 1 |
| 3 |              | 21 | 56       | 15         |        | 1 |
| 4 |              | 17 | 67       | 9          |        | 1 |
| X |              |    |          |            |        | У |
|   | numpy arrays |    |          |            |        |   |

```
KNeighborsClassifier

n_neighbors
Metric

__init__()
fit()
predict()
kneighbors()
```

kNN

## Classifiers implement the Estimator API

# KNeighborsClassifier n\_neighbors Metric \_\_init\_\_() fit() predict() kneighbors()

```
DecisionTreeClassifier

...

__init__()
fit()
predict()
...
```

```
LogisticRegression

...

__init__()
fit()
predict()
...
```

```
tree = DecisionTreeClassifier()
tree.fit(X,y)
tree.predict(X_test)
Out[33]:
array([1, 1])
```

```
lr = LogisticRegression()
lr.fit(X,y)
lr.predict(X_test)
Out[35]:
array([0, 0])
```

```
cfrs = [kNN,tree,lr]
for cfr in cfrs:
    cfr.fit(X,y)
    print(cfr.predict(X_test))

[1 0]
[1 1]
[0 0]
```

### **Polymorphism**

Estimators always have fit() and predict() methods

### **Preprocessors Implement the Transform API**

#### ■ fit and transform methods

```
from sklearn import preprocessing
scaler = preprocessing.StandardScaler().fit(X)
X scaled = scaler.transform(X)
X test scaled = scaler.transform(X test)
X test scaled
Out[49]:
array([-1.59094327, -0.05406252, -0.79537086],
       [-1.59094327, -0.10040182, -0.37117307]])
mm scaler = preprocessing.MinMaxScaler()
mm scaler.fit(X)
X test scaled = mm scaler.transform(X test)
X test scaled
Out[53]:
array([[0.125 , 0.6875 , 0.17241379],
       [0.125]
                  , 0.675 , 0.3103448311)
```

