2_performacne_metrics_for_classification

May 27, 2019

0.1 Confusion Matrix

```
In [ ]: from sklearn.metrics import confusion_matrix
        y_true = [1, 0, 1, 1, 0, 1]
        y_pred = [0, 0, 1, 1, 0, 1]
        confusion_matrix(y_true, y_pred)
```

0.2 Accuracy

```
In [ ]: import numpy as np
        from sklearn.metrics import accuracy_score
        y_pred = np.array([0, 1, 1, 0])
        y_true = np.array([0, 1, 0, 0])
```

```
In [ ]: sum(y_true == y_pred) / len(y_true)
```

```
In [ ]: accuracy_score(y_true, y_pred)
```

```
In [ ]: accuracy_score(y_true, y_pred, normalize=False)
```

0.3 Precision

```
In [ ]: from sklearn.metrics import precision_score
```

```
In [ ]: y_pred = np.array([0, 1, 1, 0])
        y_true = np.array([0, 1, 0, 0])
```

```
In [ ]: sum((y_pred == 1) & (y_pred == y_true)) / sum(y_pred)
```

```
In [ ]: precision_score(y_true, y_pred)
```

0.4 Recall

```
In [ ]: from sklearn.metrics import recall_score
```

```
In [ ]: y_pred = np.array([0, 1, 1, 0])
        y_true = np.array([0, 1, 0, 1])
```

```
In [ ]: recall_score(y_true, y_pred)
```

0.5 F1

```
In [ ]: from sklearn.metrics import f1_score
        y_pred = np.array([0, 1, 1, 0])
        y_true = np.array([0, 1, 0, 0])

In [ ]: pre = precision_score(y_true, y_pred)
        rec = recall_score(y_true, y_pred)

In [ ]: 2 * (pre * rec) / (pre + rec)

In [ ]: f1_score(y_true, y_pred)
```