kNN Example

Finding k

Finding k

- We are going to split the data into training and test data sets.
- Build kNN models for values of k between 1 and 15.
- Calculate the accuracy of each model for both training data and cross validation data.
- Plot the accuracy values on a graph.

Finding k

```
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import pandas as pd

df = pd.read_csv('data/wisc_bc_data.csv')



X = df.drop(['id', 'diagnosis'], axis='columns')
y = df.diagnosis

X_train, X_test, y_train, y_test = train_test_split(X, y,
                                    stratify=y, random_state=1)
```

Overfitting

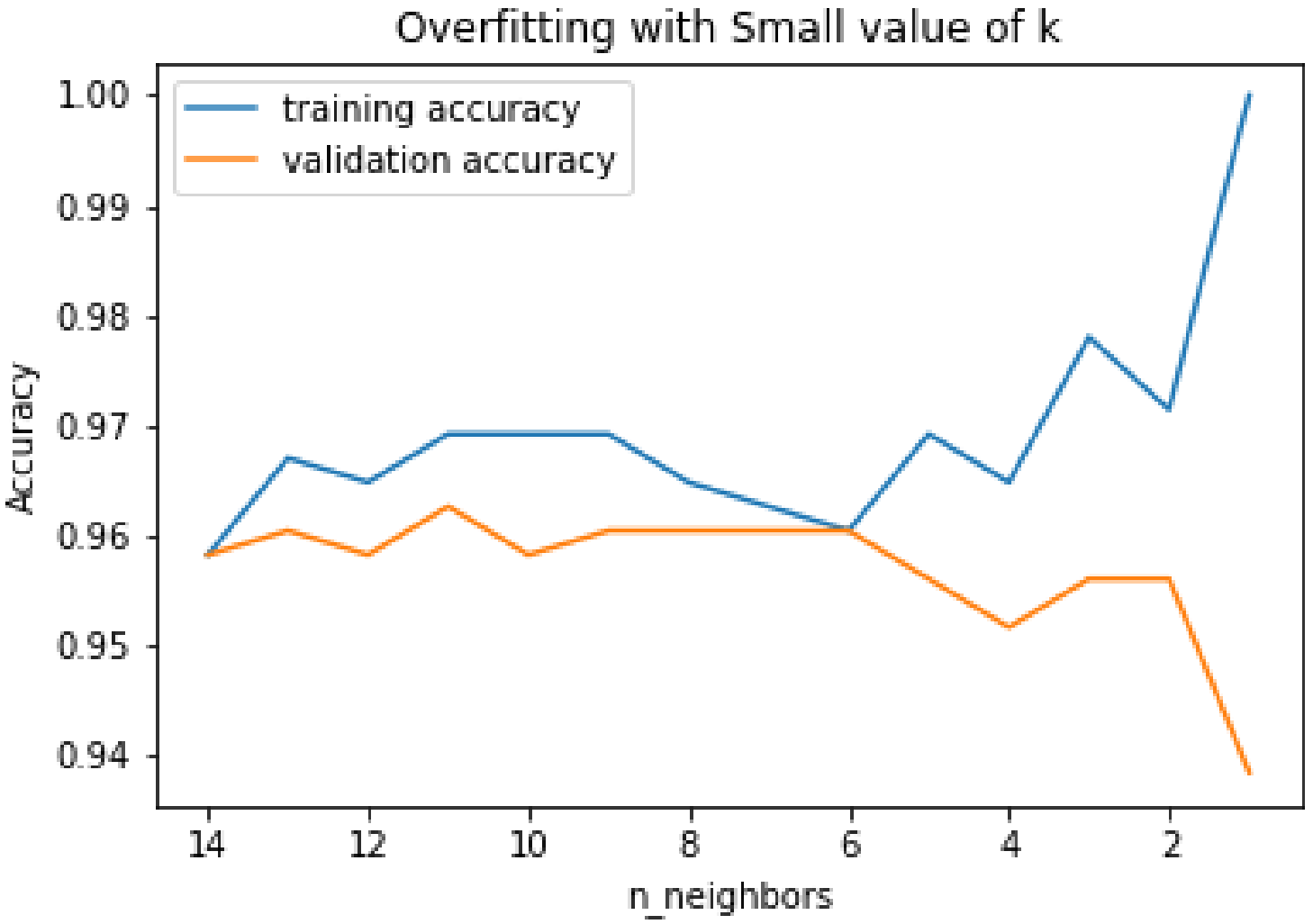
```
validation_accuracy = []
test_accuracy = []

for k in range(1,15) :
    clf = KNeighborsClassifier(n_neighbors=k)
    clf.fit(X_train, y_train)
    training_accuracy.append(clf.score(X_train, y_train))
    // find and add validation accuracy to validation_accuracy
```

Plotting

```
plt.plot(range(1,15), training_accuracy , label="training accuracy")
plt.plot(range(1,15), validation_accuracy, label="validation accuracy")
plt.xlabel("n_neighbors")
plt.ylabel("Accuracy")
plt.legend()
plt.title('Overfitting with Small value of k')
ax = plt.gca()
ax.invert_xaxis()
plt.savefig('plots/overfitting.png')
```

Methods Used

- ax = plt.gca(); ax.invert_xaxis() is used to invert the x axis.


Overfitting with Small value of k
training accuracy
validation accuracy
Accuracy
1.00
0.99
0.98
0.97
0.96
0.95
0.94
14
12
10
8
6
4
2
n_neighbors

Overfitting

- Higher values of k correspond to simpler models.
- Lower values of k to more complicated models.
- See the following plots taken from Muller.
- (This is for another data set.)
- Higher values for k give a smoother decision boundary.
-

Decision Boundaries

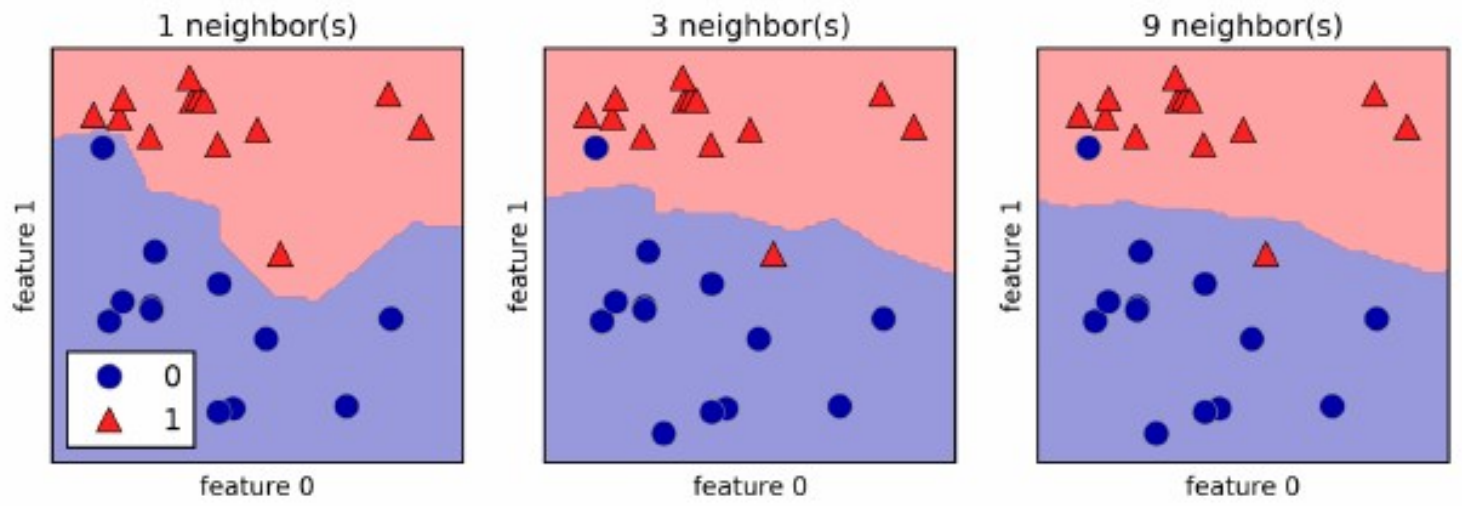


Figure 2-6. Decision boundaries created by the nearest neighbors model for different values of n_neighbors

Overfitting

- This means that we are likely to overfit by using a value of k that is too small.
- Notice from the graph that a value of k equal to 1 gives training accuracy of 100%.
- This makes sense as each data point in the training data is classified by looking at the nearest point which is itself.
-

Overfitting

- The graph above is a classic example of overfitting.
- Once k drops below 6, the test accuracy starts to drop even though the training accuracy still increases.
- It is the test accuracy we are primarily interested in.
- The value of k we should choose above is 6.

Meta-parameters

- k is an example of a meta-parameter.
- Its value needs to be fixed before we have an actual model.
- Yet nothing in the training data allows us to choose the value of the meta-parameter.
- It can only be determined by looking at performance on a validation data set.