CAC -1 Machine Learning

Mishan Regmi 1841030 5BCA -A (Fast Track)

Question Number 1

Compare and Contrast – KNN, Naïve Bayes, Decision tree classification

ANS

KNN

1. Comparing KNN with other, we choose KNN if having a conditional independence will bring negative effect on classification. And Naive Bayes can suffer from zero probability problem when conditional probability equals zero and fail to produce a valid prediction. So KNN become an easy choice in this.

2. The contrast of KNN over Naive Bayes is it can work on more boundary like linear, elliptic, or parabolic in which Naive Bayes work.

Naive Bayes

1. Comparing Naive Bayes which is a linear classification tend to work much better and faster than other classification on big data whereas KNN work slower because it requires more classification. So Naive Bayes is more use of we need speed. But KNN have more accuracy when we are not considering the factor speed. and error rate are also less.

2. Naive Bayes offers two hyperparameters to tune for smoothing. A hyperparameter is a prior parameter that are tuned on the training set to optimize it but on other side KNN have only one option for tuning.

Decision Tree

1. Of the three methods, decision trees are the easiest to explain and understand. Most people understand hierarchical trees, and the availability of a clear diagram can help you to communicate your results. Conversely, the underlying mathematics behind Bayes Theorem can be very challenging to understand for the layperson. K-NN meets somewhere in the middle; Theoretically, you could reduce the K-N

2. Decision trees have easy to use features to identify the most significant dimensions, handle missing values, and deal with outliers.

Question Number 2 & 3

```
import NumPy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split, cro
ss_val_score
from sklearn.metrics import precision_recall_fscore_suppo
rt
from sklearn.metrics import confusion_matrix , accuracy_s
core, classification_report
from sklearn import preprocessing
from sklearn import metrics

df=pd.read_csv("mushroom.csv")

df.head()

x=df.drop('class',axis = 1)
y=df['class']

cols=['cap-shape','cap-surface','cap-
color','bruises','odor','gill-attachment','gill-
spacing','gill-size','gill-color','stalk-shape', 'stalk-
root',
      'stalk-surface-above-ring','stalk-surface-below-
ring', 'stalk-color-above-ring','stalk-color-below-
ring', 'veil-type', 'veil-color',
      'ring-number','ring-type', 'spore-print-
color', 'population', 'habitat']
```

```
# Convert the data into numeric form
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
for col in cols:
     x[col]=le.fit_transform(x[col])
y=le.fit_transform(y)
x.head()


# Question 2:  K=1 case for KNN classification leads to o
ver fitting. Demonstrate it using a suitable dataset and
sample program.


xk_train,xk_test,yk_train,yk_test= train_test_split(x,y,t
est_size=0.3, random_state=42)
k = KNeighborsClassifier(n_neighbors=1)
k.fit(xk_train, yk_train)

yk_pred = k.predict(xk_test)
yk_pred

print("Testing Accuracy:",k.score(xk_test, yk_test)*100)
print("Training Accuracy:",k.score(xk_train,yk_train)*100
)

# As we can see that train accuracy is 100% so model is g
ood at training dataset
# but on test data the accuracy drops to 68.8%. at it is
leading towards overfitting.


# # Question 3:  Explain the suitability of F measure as
accuracy metric for class imbalanced data with an example
.
```

```
data=pd.read_csv("diabetes.csv")

data.head()

data['Outcome'].value_counts()

data.shape

x=data.drop('Outcome',axis = 1)
y=data['Outcome']

x_train_ib,x_test_ib,y_train_ib,y_test_ib=train_test_spli
t(x,y,test_size=0.3, random_state=42)

knn_ib=KNeighborsClassifier()
knn_ib.fit(x_train_ib,y_train_ib)
y_predict_ib=knn_ib.predict(x_test_ib)
y_predict_ib

knn_ib.score(x_test_ib,y_test_ib)

print(classification_report(y_test_ib,knn_ib.predict(x_te
st_ib)))

# When we need to find FP and FN we use F measure over ac
curacy and above the F1 score is good.
```