In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import KFold
from sklearn.model_selection import train_test_split,cross_val_scor
from sklearn.model_selection import cross_val_score
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score,classification_report,co
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import GridSearchCV
```

In [2]:

```
zoo_data = pd.read_csv("Zoo.csv")
zoo_data
zoo_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 101 entries, 0 to 100
Data columns (total 18 columns):
 #   Column       Non-Null Count  Dtype 
---  ------       --------------  ----- 
 0   animal name  101 non-null    object
 1   hair         101 non-null    int64 
 2   feathers     101 non-null    int64 
 3   eggs         101 non-null    int64 
 4   milk         101 non-null    int64 
 5   airborne     101 non-null    int64 
 6   aquatic      101 non-null    int64 
 7   predator     101 non-null    int64 
 8   toothed      101 non-null    int64 
 9   backbone     101 non-null    int64 
 10  breathes     101 non-null    int64 
 11  venomous     101 non-null    int64 
 12  fins         101 non-null    int64 
 13  legs         101 non-null    int64 
 14  tail         101 non-null    int64 
 15  domestic     101 non-null    int64 
 16  catsize      101 non-null    int64 
 17  type         101 non-null    int64 
dtypes: int64(17), object(1)
memory usage: 14.3+ KB
```

In [3]:
```
zoo_data.dtypes
```

Out[3]:
```
animal name    object
hair            int64
feathers        int64
eggs            int64
milk            int64
airborne        int64
aquatic         int64
predator        int64
toothed         int64
backbone        int64
breathes        int64
venomous        int64
fins            int64
legs            int64
tail            int64
domestic        int64
catsize         int64
type            int64
dtype: object
```

In [4]:
```
zoo_data.isnull().sum()
```

Out[4]:
```
animal name    0
hair           0
feathers       0
eggs           0
milk           0
airborne       0
aquatic        0
predator       0
toothed        0
backbone       0
breathes       0
venomous       0
fins           0
legs           0
tail           0
domestic       0
catsize        0
type           0
dtype: int64
```

In [5]:
```
zoo_data.duplicated().sum()
```

Out[5]:
```
0
```

In [6]:

```
zoo_data.describe()
```

Out[6]:

	hair	feathers	eggs	milk	airborne	aquatic	predator
count	101.000000	101.000000	101.000000	101.000000	101.000000	101.000000	101.000000
mean	0.425743	0.198020	0.584158	0.405941	0.237624	0.356436	0.554455
std	0.496921	0.400495	0.495325	0.493522	0.427750	0.481335	0.499505
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000
75%	1.000000	0.000000	1.000000	1.000000	0.000000	1.000000	1.000000
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

In [7]:

```
zoo_data['type'].unique()
```

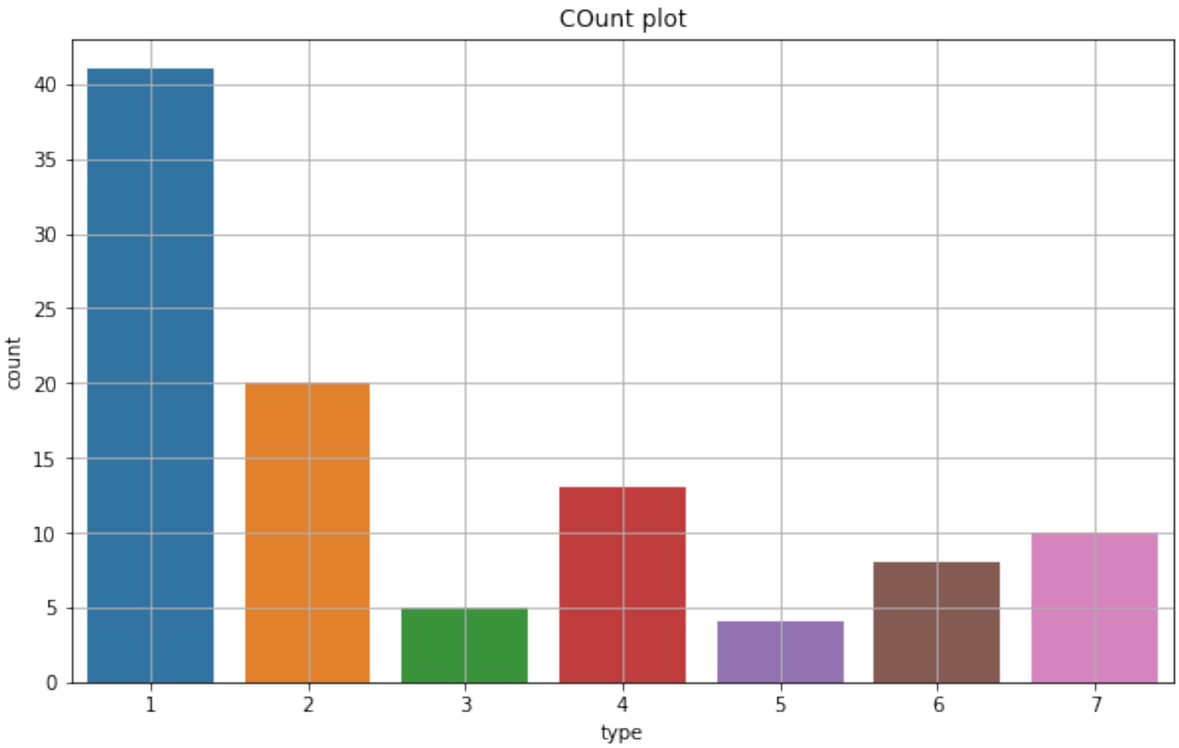
Out[7]:

```
array([1, 4, 2, 7, 6, 5, 3])
```

In [8]:

```
plt.figure(figsize=(10,6))
sns.countplot(zoo_data['type'])
plt.title('COunt plot')
plt.grid(True)
plt.show()
```

```
/opt/anaconda3/lib/python3.9/site-packages/seaborn/_decorators.py:
36: FutureWarning: Pass the following variable as a keyword arg: x
. From version 0.12, the only valid positional argument will be `d
ata`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
```



In [9]:

```
zoo_data.drop('animal name',axis=1,inplace=True)
```

In [10]:

```
zoo_data.head()
```

Out[10]:

	hair	feathers	eggs	milk	airborne	aquatic	predator	toothed	backbone	breathes	ve
0	1	0	0	1	0	0	1	1	1	1	
1	1	0	0	1	0	0	0	1	1	1	
2	0	0	1	0	0	1	1	1	1	0	
3	1	0	0	1	0	0	1	1	1	1	
4	1	0	0	1	0	0	1	1	1	1	

In [11]:
```
_data.drop('type',axis=1)
_data[['type']]
in,X_test,y_train,y_test = train_test_split(X,y,test_size=0.20,rando
('X_train_shape :',X_train.shape , '\ny_train_shape :',y_train.shape
('X_test_shape :',X_test.shape , '\ny_test_shape :',y_test.shape)
```

```
X_train_shape : (80, 16)
y_train_shape : (80, 1)
X_test_shape : (21, 16)
y_test_shape : (21, 1)
```

In [12]:
```
model = KNeighborsClassifier(n_neighbors=1)
model.fit(X_train,y_train)
```

```
/opt/anaconda3/lib/python3.9/site-packages/sklearn/neighbors/_clas
sification.py:215: DataConversionWarning: A column-vector y was pa
ssed when a 1d array was expected. Please change the shape of y to
(n_samples,), for example using ravel().
  return self._fit(X, y)
```

Out[12]:
```
▾        KNeighborsClassifier

KNeighborsClassifier(n_neighbors=1)
```

In [13]:
```
pred_y= model.predict(X_train)
```

In [14]:
```
accuracy_score(y_train,pred_y)
```

Out[14]:
```
1.0
```

In [15]:
```
confusion_matrix(y_train,pred_y)
```

Out[15]:
```
array([[34,  0,  0,  0,  0,  0,  0],
       [ 0, 17,  0,  0,  0,  0,  0],
       [ 0,  0,  4,  0,  0,  0,  0],
       [ 0,  0,  0,  9,  0,  0,  0],
       [ 0,  0,  0,  0,  3,  0,  0],
       [ 0,  0,  0,  0,  0,  6,  0],
       [ 0,  0,  0,  0,  0,  0,  7]])
```

```
In [16]: print(classification_report(y_train,pred_y))
```

```
              precision    recall  f1-score   support

           1       1.00      1.00      1.00        34
           2       1.00      1.00      1.00        17
           3       1.00      1.00      1.00         4
           4       1.00      1.00      1.00         9
           5       1.00      1.00      1.00         3
           6       1.00      1.00      1.00         6
           7       1.00      1.00      1.00         7

    accuracy                           1.00        80
   macro avg       1.00      1.00      1.00        80
weighted avg       1.00      1.00      1.00        80
```

```
In [17]: y_pred=model.predict(X_test)
```

```
In [18]: accuracy_score(y_test,y_pred)
```

```
Out[18]: 0.9523809523809523
```

```
In [19]: confusion_matrix(y_test,y_pred)
```

```
Out[19]: array([[7, 0, 0, 0, 0, 0, 0],
                [0, 3, 0, 0, 0, 0, 0],
                [0, 0, 1, 0, 0, 0, 0],
                [0, 0, 0, 4, 0, 0, 0],
                [0, 0, 0, 0, 1, 0, 0],
                [0, 0, 0, 0, 0, 2, 0],
                [0, 0, 0, 0, 1, 0, 2]])
```

```
In [20]: print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           1       1.00      1.00      1.00         7
           2       1.00      1.00      1.00         3
           3       1.00      1.00      1.00         1
           4       1.00      1.00      1.00         4
           5       0.50      1.00      0.67         1
           6       1.00      1.00      1.00         2
           7       1.00      0.67      0.80         3

    accuracy                           0.95        21
   macro avg       0.93      0.95      0.92        21
weighted avg       0.98      0.95      0.96        21
```

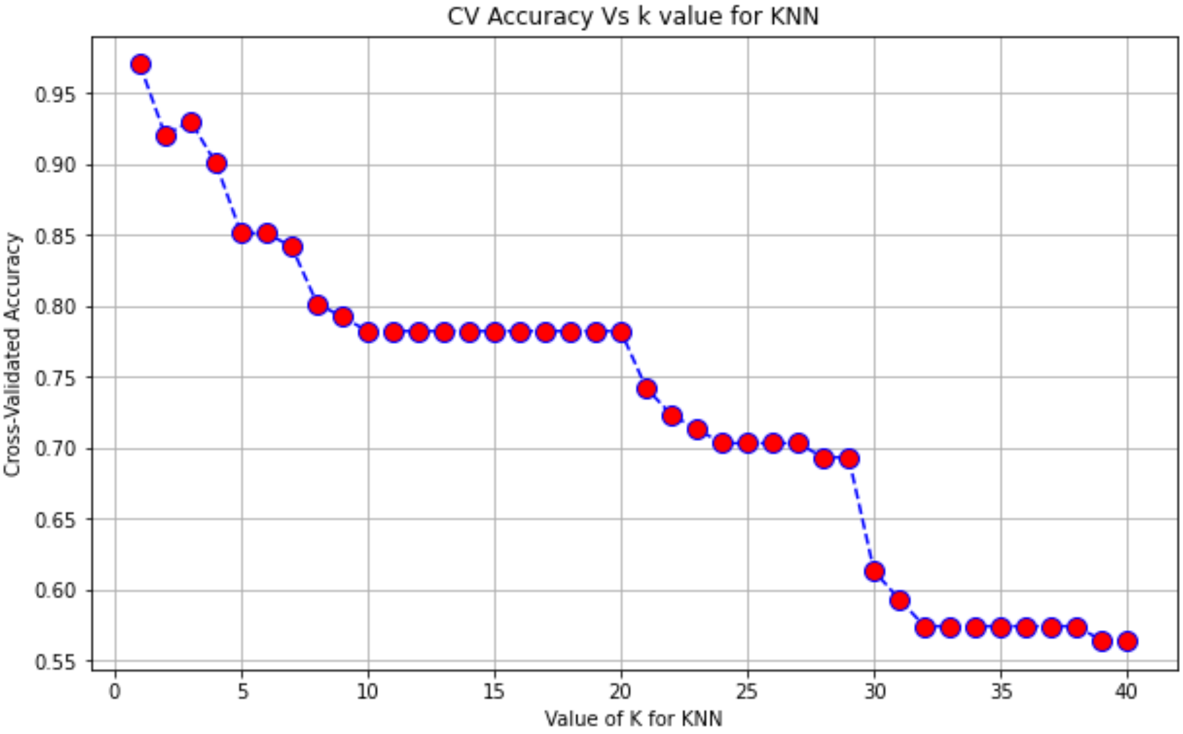
In [21]:

```
import matplotlib.pyplot as plt
%matplotlib inline
# choose k between 1 to 41
k_range = range(1, 41)
k_scores = []
# use iteration to caclulator different k in models, then return th
for k in k_range:
    knn = KNeighborsClassifier(n_neighbors=k)
    scores = cross_val_score(knn, X, y, cv=5)
    k_scores.append(scores.mean())
```

```
/opt/anaconda3/lib/python3.9/site-packages/sklearn/neighbors/_clas
sification.py:215: DataConversionWarning: A column-vector y was pa
ssed when a 1d array was expected. Please change the shape of y to
(n_samples,), for example using ravel().
  return self._fit(X, y)
/opt/anaconda3/lib/python3.9/site-packages/sklearn/neighbors/_clas
sification.py:215: DataConversionWarning: A column-vector y was pa
ssed when a 1d array was expected. Please change the shape of y to
(n_samples,), for example using ravel().
  return self._fit(X, y)
/opt/anaconda3/lib/python3.9/site-packages/sklearn/neighbors/_clas
sification.py:215: DataConversionWarning: A column-vector y was pa
ssed when a 1d array was expected. Please change the shape of y to
(n_samples,), for example using ravel().
  return self._fit(X, y)
/opt/anaconda3/lib/python3.9/site-packages/sklearn/neighbors/_clas
sification.py:215: DataConversionWarning: A column-vector y was pa
ssed when a 1d array was expected. Please change the shape of y to
(n_samples,), for example using ravel().
  return self._fit(X, y)
```

In [22]:

```
# plot to see clearly
plt.figure(figsize=(10,6))
plt.plot(k_range, k_scores,color='blue',linestyle='dashed',marker='
plt.grid(True)
plt.title('CV Accuracy Vs k value for KNN')
plt.xlabel('Value of K for KNN')
plt.ylabel('Cross-Validated Accuracy')
plt.show()
```



In [ ]: