

abalone-categorical

In [1]:

```
import numpy as np
import pandas as pd
```

In [2]:

```
dataset = pd.read_csv(r"C:\Users\kotha\Downloads\Abalone_data.csv")
```

In [3]:

```
dataset
```

Out[3]:

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_we
0	M	70	53	18	45.1	19.9	9.7	.
1	F	106	84	27	135.4	51.3	28.3	.
2	M	88	73	25	103.2	43.1	22.8	.
3	I	66	51	16	41.0	17.9	7.9	.
4	I	85	60	19	70.3	28.2	15.5	.
...
4171	F	113	90	33	177.4	74.0	47.8	.
4172	M	118	88	27	193.2	87.8	42.9	.
4173	M	120	95	41	235.2	105.1	57.5	.
4174	F	125	97	30	218.9	106.2	52.2	.
4175	M	142	111	39	389.7	189.1	75.3	.

4176 rows × 9 columns



In [4]:

```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4176 entries, 0 to 4175
Data columns (total 9 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Sex                    4176 non-null   object
1   Length                 4176 non-null   int64
2   Diameter               4176 non-null   int64
3   Height                 4176 non-null   int64
4   Whole_weight           4176 non-null   float64
5   Shucked_weight         4176 non-null   float64
6   Viscera_weight         4176 non-null   float64
7   Shell_weight           4176 non-null   float64
8   Rings                 4176 non-null   int64
dtypes: float64(4), int64(4), object(1)
memory usage: 293.8+ KB
```

In [5]:

```
dataset["Sex"].unique()
```

Out[5]:

```
array(['M', 'F', 'I'], dtype=object)
```

In [6]:

```
dataset.isnull().sum()
```

Out[6]:

```
Sex                0
Length             0
Diameter           0
Height             0
Whole_weight       0
Shucked_weight     0
Viscera_weight     0
Shell_weight       0
Rings              0
dtype: int64
```

In [7]:

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
dataset["Sex"] = le.fit_transform(dataset["Sex"])
```

In [8]:

```
dataset.head(5)
```

Out[8]:

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight
0	2	70	53	18	45.1	19.9	9.7	14.0
1	0	106	84	27	135.4	51.3	28.3	42.0
2	2	88	73	25	103.2	43.1	22.8	31.0
3	1	66	51	16	41.0	17.9	7.9	11.0
4	1	85	60	19	70.3	28.2	15.5	24.0

In [9]:

```
x = dataset.iloc[:,1:9].values  
y = dataset.iloc[:,0:1].values
```

In [10]:

```
x
```

Out[10]:

```
array([[ 70. ,  53. ,  18. , ...,   9.7,  14. ,   7. ],  
       [106. ,  84. ,  27. , ...,  28.3,  42. ,   9. ],  
       [ 88. ,  73. ,  25. , ...,  22.8,  31. ,  10. ],  
       ...,  
       [120. ,  95. ,  41. , ...,  57.5,  61.6,   9. ],  
       [125. ,  97. ,  30. , ...,  52.2,  59.2,  10. ],  
       [142. , 111. ,  39. , ...,  75.3,  99. ,  12. ]])
```

In [11]:

```
y
```

Out[11]:

```
array([[2],  
       [0],  
       [2],  
       ...,  
       [2],  
       [0],  
       [2]])
```

In [12]:

```
from sklearn.preprocessing import OneHotEncoder
one = OneHotEncoder()
z = one.fit_transform(y[:,0:1]).toarray()
y = np.delete(y,0,axis=1)
y = np.concatenate((z,y),axis=1)
```

In [13]:

```
y.shape
```

Out[13]:

```
(4176, 3)
```

In [14]:

```
x.shape
```

Out[14]:

```
(4176, 8)
```

In [15]:

```
y
```

Out[15]:

```
array([[0., 0., 1.],
       [1., 0., 0.],
       [0., 0., 1.],
       ...,
       [0., 0., 1.],
       [1., 0., 0.],
       [0., 0., 1.]])
```

In [16]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.2, random_state = 0)
```

In [17]:

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.transform(x_test)
```

In [18]:

```
from keras.models import Sequential
from keras.layers import Dense
```

Using TensorFlow backend.

C:\Users\kotha\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:516: FutureWarning: Passing (type, 1) or '1type' as a synonym of type i
s deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.

```
_np_qint8 = np.dtype [("qint8", np.int8, 1)]
```

C:\Users\kotha\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:517: FutureWarning: Passing (type, 1) or '1type' as a synonym of type i
s deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.

```
_np_quint8 = np.dtype [("quint8", np.uint8, 1)]
```

C:\Users\kotha\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:518: FutureWarning: Passing (type, 1) or '1type' as a synonym of type i
s deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.

```
_np_qint16 = np.dtype [("qint16", np.int16, 1)]
```

C:\Users\kotha\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:519: FutureWarning: Passing (type, 1) or '1type' as a synonym of type i
s deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.

```
_np_quint16 = np.dtype [("quint16", np.uint16, 1)]
```

C:\Users\kotha\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:520: FutureWarning: Passing (type, 1) or '1type' as a synonym of type i
s deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.

```
_np_qint32 = np.dtype [("qint32", np.int32, 1)]
```

C:\Users\kotha\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:525: FutureWarning: Passing (type, 1) or '1type' as a synonym of type i
s deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.

```
np_resource = np.dtype [("resource", np.ubyte, 1)]
```

C:\Users\kotha\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stu
b\dtypes.py:541: FutureWarning: Passing (type, 1) or '1type' as a synonym of
type is deprecated; in a future version of numpy, it will be understood as
(type, (1,)) / '(1,)type'.

```
_np_qint8 = np.dtype [("qint8", np.int8, 1)]
```

C:\Users\kotha\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stu
b\dtypes.py:542: FutureWarning: Passing (type, 1) or '1type' as a synonym of
type is deprecated; in a future version of numpy, it will be understood as
(type, (1,)) / '(1,)type'.

```
_np_quint8 = np.dtype [("quint8", np.uint8, 1)]
```

C:\Users\kotha\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stu
b\dtypes.py:543: FutureWarning: Passing (type, 1) or '1type' as a synonym of
type is deprecated; in a future version of numpy, it will be understood as
(type, (1,)) / '(1,)type'.

```
_np_qint16 = np.dtype [("qint16", np.int16, 1)]
```

C:\Users\kotha\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stu
b\dtypes.py:544: FutureWarning: Passing (type, 1) or '1type' as a synonym of
type is deprecated; in a future version of numpy, it will be understood as
(type, (1,)) / '(1,)type'.

```
_np_quint16 = np.dtype [("quint16", np.uint16, 1)]
```

C:\Users\kotha\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stu
b\dtypes.py:545: FutureWarning: Passing (type, 1) or '1type' as a synonym of
type is deprecated; in a future version of numpy, it will be understood as
(type, (1,)) / '(1,)type'.

```
_np_qint32 = np.dtype [("qint32", np.int32, 1)]
```

```
C:\Users\kotha\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:550: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np_resource = np.dtype [("resource", np.ubyte, 1)])
```

In [19]:

```
model = Sequential()
```

```
WARNING:tensorflow:From C:\Users\kotha\anaconda3\lib\site-packages\keras\backend\tensorflow_backend.py:74: The name tf.get_default_graph is deprecated. Please use tf.compat.v1.get_default_graph instead.
```

In [20]:

```
x.shape
```

Out[20]:

```
(4176, 8)
```

In [21]:

```
model.add(Dense(units = 8 ,activation = "relu", init = "random_uniform"))
```

```
C:\Users\kotha\anaconda3\lib\site-packages\ipykernel_launcher.py:1: UserWarning: Update your `Dense` call to the Keras 2 API: `Dense(units=8, activation="relu", kernel_initializer="random_uniform")`
  """Entry point for launching an IPython kernel.
```

In [22]:

```
model.add(Dense(units = 16 ,activation = "relu", init = "random_uniform"))
```

```
C:\Users\kotha\anaconda3\lib\site-packages\ipykernel_launcher.py:1: UserWarning: Update your `Dense` call to the Keras 2 API: `Dense(units=16, activation="relu", kernel_initializer="random_uniform")`
  """Entry point for launching an IPython kernel.
```

In [23]:

```
model.add(Dense(units = 32 ,activation = "relu", init = "random_uniform"))
```

```
C:\Users\kotha\anaconda3\lib\site-packages\ipykernel_launcher.py:1: UserWarning: Update your `Dense` call to the Keras 2 API: `Dense(units=32, activation="relu", kernel_initializer="random_uniform")`
  """Entry point for launching an IPython kernel.
```

In [24]:

```
model.add(Dense(units = 3 ,activation = "softmax", init = "random_uniform"))
```

```
C:\Users\kotha\anaconda3\lib\site-packages\ipykernel_launcher.py:1: UserWarning: Update your `Dense` call to the Keras 2 API: `Dense(units=3, activation="softmax", kernel_initializer="random_uniform")`
  """Entry point for launching an IPython kernel.
```

In [25]:

```
model.compile(optimizer = "adam", loss = "categorical_crossentropy", metrics = ["accuracy"])
```

WARNING:tensorflow:From C:\Users\kotha\anaconda3\lib\site-packages\keras\optimizers.py:790: The name tf.train.Optimizer is deprecated. Please use tf.compat.v1.train.Optimizer instead.

In [26]:

```
model.fit(x_train,y_train, epochs = 10000, batch_size = 32)
```

WARNING:tensorflow:From C:\Users\kotha\anaconda3\lib\site-packages\keras\backend\tensorflow_backend.py:517: The name tf.placeholder is deprecated. Please use tf.compat.v1.placeholder instead.

WARNING:tensorflow:From C:\Users\kotha\anaconda3\lib\site-packages\keras\backend\tensorflow_backend.py:4138: The name tf.random_uniform is deprecated. Please use tf.random.uniform instead.

WARNING:tensorflow:From C:\Users\kotha\anaconda3\lib\site-packages\keras\backend\tensorflow_backend.py:3295: The name tf.log is deprecated. Please use tf.math.log instead.

WARNING:tensorflow:From C:\Users\kotha\anaconda3\lib\site-packages\tensorflow\python\ops\math_grad.py:1250: add_dispatch_support.<locals>.wrapper (from tensorflow.python.ops.array_ops) is deprecated and will be removed in a future version.

Instructions for updating:

Use tf.where in 2.0, which has the same broadcast rule as np.where

WARNING:tensorflow:From C:\Users\kotha\anaconda3\lib\site-packages\keras\backend\tensorflow_backend.py:3295: The name tf.log is deprecated. Please use tf.math.log instead.

In [27]:

```
y_pred = model.predict(x_test)
```

In [28]:

```
y_pred
```

Out[28]:

```
array([[5.32263935e-01, 6.03636960e-03, 4.61699724e-01],
       [1.95544692e-11, 9.85588610e-01, 1.44113684e-02],
       [4.45253932e-07, 9.56936300e-01, 4.30632606e-02],
       ...,
       [1.89774390e-02, 9.66137767e-01, 1.48847345e-02],
       [9.00125980e-01, 2.89316238e-10, 9.98740047e-02],
       [9.79598224e-01, 7.77429250e-07, 2.04009749e-02]], dtype=float32)
```

In [29]:

```
y_test
```

Out[29]:

```
array([[0., 0., 1.],
       [0., 1., 0.],
       [0., 1., 0.],
       ...,
       [0., 1., 0.],
       [1., 0., 0.],
       [1., 0., 0.]])
```

In [30]:

```
yp = model.predict(sc.transform(np.array([[140,116,41,426,148.3,98,116,20]])))
```

In [31]:

```
yp
```

Out[31]:

```
array([[1.1121402e-02, 3.8313380e-04, 9.8849547e-01]], dtype=float32)
```

In [32]:

```
ypc = model.predict_classes(sc.transform(np.array([[140,116,41,426,148.3,98,116,20]])))
```

In [33]:

```
ypc
```

Out[33]:

```
array([2], dtype=int64)
```

In [34]:

```
p = ypc[0]
```

In [35]:

```
p
```

Out[35]:

```
2
```

In [36]:

```
index = ['F', 'I', 'M']
```

In [37]:

```
print(index[p])
```

```
M
```


In []: