

Hands_on_Activity_7_1

Technological Institute of the
Philippines

Course Code:

Code Title:

Summer

Quezon City - Computer Engineering

CPE 019

Emerging Technologies in CpE 2

AY 2024 - 2025

Hands-on Activity 7.1

Name

Section

Date Performed:

Date Submitted:

Instructor:

Classifications and Regression

Calvadores, Kelly Joseph

CPE32S1

June 28, 2024

June , 2024

Engr. Roman M. Richard

Choose any dataset applicable to the classification problem, and also, choose any dataset applicable to the regression problem.¶

Explain your datasets and the problem being addressed.¶

- The problem that is currently being addressed is the quality of the 2 wines, the goal for this, is to model the wine quality based on the tests that has been made. To approach the quality of the wine based on the physicochemical properties that is given, the model can help to predict the aiding quality control of the wine and decision making in the wine industry

For classification, do the following:¶

In [2]:

```
!pip install scikeras
```

```
Collecting scikeras
```

```
  Downloading scikeras-0.13.0-py3-none-any.whl (26 kB)
```

```
Collecting keras>=3.2.0 (from scikeras)
```

```
  Downloading keras-3.4.1-py3-none-any.whl (1.1 MB)
```

1.1/1.1 MB 6.9 MB/s eta 0:00:00
Collecting scikit-learn>=1.4.2 (from scikeras)
 Downloading
scikit_learn-1.5.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
(13.3 MB)

13.3/13.3 MB 50.9 MB/s eta 0:00:00
Requirement already satisfied: absl-py in
/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras) (1.4.0)
Requirement already satisfied: numpy in
/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras)
(1.25.2)
Requirement already satisfied: rich in
/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras)
(13.7.1)
Collecting namex (from keras>=3.2.0->scikeras)
 Downloading namex-0.0.8-py3-none-any.whl (5.8 kB)
Requirement already satisfied: h5py in
/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras) (3.9.0)
Collecting optree (from keras>=3.2.0->scikeras)
 Downloading
optree-0.11.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (311
kB)

311.2/311.2 kB 28.3 MB/s eta 0:00:00
Requirement already satisfied: ml-dtypes in
/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras) (0.2.0)
Requirement already satisfied: packaging in
/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras) (24.1)
Requirement already satisfied: scipy>=1.6.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn>=1.4.2->scikeras)
(1.11.4)
Requirement already satisfied: joblib>=1.2.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn>=1.4.2->scikeras)
(1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn>=1.4.2->scikeras)
(3.5.0)
Requirement already satisfied: typing-extensions>=4.0.0 in
/usr/local/lib/python3.10/dist-packages (from optree->keras>=3.2.0->scikeras)
(4.12.2)
Requirement already satisfied: markdown-it-py>=2.2.0 in
/usr/local/lib/python3.10/dist-packages (from rich->keras>=3.2.0->scikeras)
(3.0.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in
/usr/local/lib/python3.10/dist-packages (from rich->keras>=3.2.0->scikeras)
(2.16.1)
Requirement already satisfied: mdurl~=0.1 in

```

/usr/local/lib/python3.10/dist-packages (from
markdown-it-py>=2.2.0->rich->keras>=3.2.0->scikeras) (0.1.2)
Installing collected packages: namex, optree, scikit-learn, keras, scikeras
Attempting uninstall: scikit-learn
  Found existing installation: scikit-learn 1.2.2
  Uninstalling scikit-learn-1.2.2:
    Successfully uninstalled scikit-learn-1.2.2
Attempting uninstall: keras
  Found existing installation: keras 2.15.0
  Uninstalling keras-2.15.0:
    Successfully uninstalled keras-2.15.0
ERROR: pip's dependency resolver does not currently take into account all the
packages that are installed. This behaviour is the source of the following
dependency conflicts.
tensorflow 2.15.0 requires keras<2.16,>=2.15.0, but you have keras 3.4.1
which is incompatible.
Successfully installed keras-3.4.1 namex-0.0.8 optree-0.11.0 scikeras-0.13.0
scikit-learn-1.5.0

```

In [1]:

```

import pandas as pd
from keras.models import Sequential
from keras.layers import Dense
from scikeras.wrappers import KerasClassifier
from tensorflow.keras.utils import to_categorical
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold
from sklearn.preprocessing import LabelEncoder
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split

```

Resource: <https://archive.ics.uci.edu/dataset/186/wine+quality>

In [9]:

```

CData = pd.read_csv('/content/drive/MyDrive/CPE 019 (Retake)/HOA
7.1/WineQualityData/winequality-white.csv', sep = ';')
CData

```

Out[9]:

		fixed	volat	residual	free	total						
		acidity	ile	citric	sulfur	sulfur	density	pH	sulfate	alcohol	quality	
0	7.0	0.27	0.36	20.7	0.045	170.0	1.00100	3.00	0.45	8.8	6	

1	6.3	0.30	0.34	1.6	0.049	14.0	132.0	0.99400	3.30	0.49	9.5	6
2	8.1	0.28	0.40	6.9	0.050	30.0	97.0	0.99510	3.26	0.44	10.1	6
3	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.40	9.9	6
4	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.40	9.9	6
...
4893	6.2	0.21	0.29	1.6	0.039	24.0	92.0	0.99114	3.27	0.50	11.2	6
4894	6.6	0.32	0.36	8.0	0.047	57.0	168.0	0.99490	3.15	0.46	9.6	5
4895	6.5	0.24	0.19	1.2	0.041	30.0	111.0	0.99254	2.99	0.46	9.4	6
4896	5.5	0.29	0.30	1.1	0.022	20.0	110.0	0.98869	3.34	0.38	12.8	7
4897	6.0	0.21	0.38	0.8	0.020	22.0	98.0	0.98941	3.26	0.32	11.8	6

4898 rows × 12 columns

In [10]:

CData.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 4898 entries, 0 to 4897

Data columns (total 12 columns):

#	Column	Non-Null Count	Dtype
0	fixed acidity	4898 non-null	float64
1	volatile acidity	4898 non-null	float64
2	citric acid	4898 non-null	float64
3	residual sugar	4898 non-null	float64
4	chlorides	4898 non-null	float64
5	free sulfur dioxide	4898 non-null	float64
6	total sulfur dioxide	4898 non-null	float64
7	density	4898 non-null	float64
8	pH	4898 non-null	float64
9	sulphates	4898 non-null	float64
10	alcohol	4898 non-null	float64
11	quality	4898 non-null	int64

dtypes: float64(11), int64(1)

memory usage: 459.3 KB

In [11]:

```
CData.corr()
```

Out[11]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chloride s	free sulfur dioxide	total sulfur dioxide	density	pH	sulfate	alcohol	quality
fixed acidity	1.0000	-0.22697	0.289181	0.089021	0.023086	-0.49396	0.091070	0.265331	-0.4258	-0.017143	-0.120881	-0.113663
volatile acidity	-0.22697	1.00000	-0.149472	0.064286	0.070512	-0.97012	0.089261	0.027114	-0.031915	-0.035728	0.067718	-0.194723
citric acid	0.289181	-0.149472	1.000000	0.094212	0.114364	0.094077	0.121131	0.149503	-0.163748	0.062331	-0.075729	-0.009209
residual sugar	0.089021	0.064286	0.094212	1.000000	0.088685	0.299098	0.401439	0.838966	-0.194133	-0.026664	-0.450631	-0.097577
chloride s	0.023086	0.070512	0.114364	0.088685	1.000000	0.101392	0.198910	0.257211	-0.090439	0.016763	-0.360189	-0.209934
free sulfur dioxide	-0.49396	-0.97012	0.094077	0.299098	0.101392	1.000000	0.615501	0.294210	-0.000618	0.059217	-0.250104	0.008158
total sulfur dioxide	0.091070	0.089261	0.121131	0.401439	0.198910	0.615501	1.000000	0.529881	0.002321	0.134562	-0.448892	-0.174737
density	0.265331	0.027114	0.149503	0.838966	0.257211	0.294210	0.529881	1.000000	-0.093591	0.074493	-0.780138	-0.307123
pH	-0.4258	-0.031915	-0.163748	-0.194133	-0.090439	-0.000618	0.002321	-0.093591	1.000000	0.155951	0.121432	0.099427

sulphate	-0.0	-0.0	0.06	-0.0	0.01	0.05	0.13	0.07	0.15	1.00	-0.0	0.05
	171	357	233	266	676	921	456	449	595	000	174	367
	43	28	1	64	3	7	2	3	1	0	33	8
alcohol	-0.1	0.06	-0.0	-0.4	-0.3	-0.2	-0.4	-0.7	0.12	-0.0	1.00	0.43
	208	771	757	506	601	501	488	801	143	174	000	557
	81	8	29	31	89	04	92	38	2	33	0	5
quality	-0.1	-0.1	-0.0	-0.0	-0.2	0.00	-0.1	-0.3	0.09	0.05	0.43	1.00
	136	947	092	975	099	815	747	071	942	367	557	000
	63	23	09	77	34	8	37	23	7	8	5	0

In [12]:

```
CorrMatr = CData.corr()
TargCorr = CorrMatr['quality']
AbsTarCorr = TargCorr.abs()
LowCorrFeat = AbsTarCorr[AbsTarCorr < 0.01]
print(LowCorrFeat)
```

```
citric acid          0.009209
free sulfur dioxide  0.008158
Name: quality, dtype: float64
```

In [13]:

```
CData = CData.drop(columns = LowCorrFeat.index)
CData
```

Out[13]:

	fixed acidit y	volatil e acidit y	resid ual sugar	chlori des	total sulfur dioxid e	densit y	pH	sulph ates	alcoh ol	qualit y
0	7.0	0.27	20.7	0.045	170.0	1.001 00	3.00	0.45	8.8	6
1	6.3	0.30	1.6	0.049	132.0	0.994 00	3.30	0.49	9.5	6
2	8.1	0.28	6.9	0.050	97.0	0.995 10	3.26	0.44	10.1	6
3	7.2	0.23	8.5	0.058	186.0	0.995 60	3.19	0.40	9.9	6
4	7.2	0.23	8.5	0.058	186.0	0.995 60	3.19	0.40	9.9	6
...
4893	6.2	0.21	1.6	0.039	92.0	0.991 14	3.27	0.50	11.2	6

4894	6.6	0.32	8.0	0.047	168.0	0.994	3.15	0.46	9.6	5
						90				
4895	6.5	0.24	1.2	0.041	111.0	0.992	2.99	0.46	9.4	6
						54				
4896	5.5	0.29	1.1	0.022	110.0	0.988	3.34	0.38	12.8	7
						69				
4897	6.0	0.21	0.8	0.020	98.0	0.989	3.26	0.32	11.8	6
						41				

4898 rows × 10 columns

In [21]:

CData.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4898 entries, 0 to 4897
Data columns (total 10 columns):
#   Column                Non-Null Count  Dtype
---  -
0   fixed acidity          4898 non-null   float64
1   volatile acidity       4898 non-null   float64
2   residual sugar         4898 non-null   float64
3   chlorides              4898 non-null   float64
4   total sulfur dioxide   4898 non-null   float64
5   density                4898 non-null   float64
6   pH                    4898 non-null   float64
7   sulphates              4898 non-null   float64
8   alcohol                4898 non-null   float64
9   quality                4898 non-null   int64
dtypes: float64(9), int64(1)
memory usage: 382.8 KB
```

In [14]:

```
CX = CData.iloc[:, :-1]
Cy = CData.iloc[:, -1]
```

In [15]:

```
SS = StandardScaler()
CX = SS.fit_transform(CX)
```

In [16]:

```
CX_train, CX_test, Cy_train, Cy_test = train_test_split(CX, Cy, test_size =
0.25, random_state = 123)
```

In [17]:

```

LE = LabelEncoder()
Cy_train = LE.fit_transform(Cy_train)
Cy_test = LE.fit_transform(Cy_test)
Cy_train = to_categorical(Cy_train)
Cy_test = to_categorical(Cy_test)

```

In [18]:

```
Cy_train.shape
```

Out[18]:

```
(3673, 7)
```

Create a base model¶

In [22]:

```

def Cbaseline_model():
    model = Sequential()
    model.add(Dense(32, input_dim = 9, activation = 'relu'))
    model.add(Dense(7, activation = 'softmax'))

    model.compile(loss = 'categorical_crossentropy', optimizer = 'adam',
metrics = ['accuracy'])
    return model

```

Evaluate the model with k-fold cross validation¶

In [24]:

```

CEstimator = KerasClassifier(model = Cbaseline_model, epochs = 500,
batch_size = 5000, verbose = 0)
Ckfold = KFold(n_splits = 10, shuffle = True)
CResults = cross_val_score(CEstimator, CX_train, Cy_train, cv = Ckfold)
print("Baseline: %.2f%% (%.2f%%)" % (CResults.mean()*100,
CResults.std()*100))

/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

```



```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

Baseline: 54.91% (2.19%)

Improve the accuracy of your model by applying additional hidden layers¶

In [27]:

```
def Cbaseline_model():
    model = Sequential()
    model.add(Dense(128, input_dim = 9, activation = 'relu'))
    model.add(Dense(64, activation = 'relu'))
    model.add(Dense(32, activation = 'relu'))
    model.add(Dense(7, activation = 'softmax'))
```

```

    model.compile(loss = 'categorical_crossentropy', optimizer = 'adam',
metrics = ['accuracy'])
    return model

```

In [28]:

```

CEstimator = KerasClassifier(model = Cbaseline_model, epochs = 500,
batch_size = 5000, verbose = 0)
Ckfold = KFold(n_splits = 10, shuffle = True)
CResults = cross_val_score(CEstimator, CX_train, Cy_train, cv = Ckfold)
print("Baseline: %.2f%% (%.2f%%)" % (CResults.mean()*100,
CResults.std()*100))

/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
    super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
    super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
    super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
    super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
    super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
    super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
    super().__init__(activity_regularizer=activity_regularizer, **kwargs)

```

When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

Baseline: 56.25% (2.74%)

For regression, do the following:¶

In [2]:

```
!pip install scikeras
```

Collecting scikeras

Downloading scikeras-0.13.0-py3-none-any.whl (26 kB)

Collecting keras>=3.2.0 (from scikeras)

Downloading keras-3.4.1-py3-none-any.whl (1.1 MB)

1.1/1.1 MB 6.1 MB/s eta 0:00:00

Collecting scikit-learn>=1.4.2 (from scikeras)

Downloading

scikit_learn-1.5.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
(13.3 MB)

13.3/13.3 MB 44.0 MB/s eta 0:00:00

Requirement already satisfied: absl-py in

/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras) (1.4.0)

Requirement already satisfied: numpy in

/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras)
(1.25.2)

Requirement already satisfied: rich in

/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras)
(13.7.1)

Collecting namex (from keras>=3.2.0->scikeras)

Downloading namex-0.0.8-py3-none-any.whl (5.8 kB)

Requirement already satisfied: h5py in

/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras) (3.9.0)

Collecting optree (from keras>=3.2.0->scikeras)

Downloading
optree-0.11.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (311 kB)

```
311.2/311.2 kB 30.0 MB/s eta 0:00:00
Requirement already satisfied: ml-dtypes in
/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras) (0.2.0)
Requirement already satisfied: packaging in
/usr/local/lib/python3.10/dist-packages (from keras>=3.2.0->scikeras) (24.1)
Requirement already satisfied: scipy>=1.6.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn>=1.4.2->scikeras)
(1.11.4)
Requirement already satisfied: joblib>=1.2.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn>=1.4.2->scikeras)
(1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn>=1.4.2->scikeras)
(3.5.0)
Requirement already satisfied: typing-extensions>=4.0.0 in
/usr/local/lib/python3.10/dist-packages (from optree->keras>=3.2.0->scikeras)
(4.12.2)
Requirement already satisfied: markdown-it-py>=2.2.0 in
/usr/local/lib/python3.10/dist-packages (from rich->keras>=3.2.0->scikeras)
(3.0.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in
/usr/local/lib/python3.10/dist-packages (from rich->keras>=3.2.0->scikeras)
(2.16.1)
Requirement already satisfied: mdurl~=0.1 in
/usr/local/lib/python3.10/dist-packages (from
markdown-it-py>=2.2.0->rich->keras>=3.2.0->scikeras) (0.1.2)
Installing collected packages: namex, optree, scikit-learn, keras, scikeras
  Attempting uninstall: scikit-learn
    Found existing installation: scikit-learn 1.2.2
    Uninstalling scikit-learn-1.2.2:
      Successfully uninstalled scikit-learn-1.2.2
  Attempting uninstall: keras
    Found existing installation: keras 2.15.0
    Uninstalling keras-2.15.0:
      Successfully uninstalled keras-2.15.0
ERROR: pip's dependency resolver does not currently take into account all the
packages that are installed. This behaviour is the source of the following
dependency conflicts.
tensorflow 2.15.0 requires keras<2.16,>=2.15.0, but you have keras 3.4.1
which is incompatible.
Successfully installed keras-3.4.1 namex-0.0.8 optree-0.11.0 scikeras-0.13.0
scikit-learn-1.5.0
```

In [1]:

```

import pandas as pd
from keras.models import Sequential
from keras.layers import Dense
from scikeras.wrappers import KerasRegressor
from tensorflow.keras.utils import to_categorical
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold
from sklearn.preprocessing import LabelEncoder
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split

```

In [2]:

```

RData = pd.read_csv('/content/drive/MyDrive/CPE 019 (Retake)/HOA
7.1/WineQualityData/winequality-red.csv', sep = ';')
RData

```

Out[2]:

	fixed acidi ty	volat ile acidi ty	citri c acid	resi dual suga r	chlo ride s	free sulfu r dioxi de	total sulfu r dioxi de	dens ity	pH	sulp hate s	alco hol	qual ity
0	7.4	0.70 0	0.00	1.9	0.07 6	11.0	34.0	0.99 780	3.51	0.56	9.4	5
1	7.8	0.88 0	0.00	2.6	0.09 8	25.0	67.0	0.99 680	3.20	0.68	9.8	5
2	7.8	0.76 0	0.04	2.3	0.09 2	15.0	54.0	0.99 700	3.26	0.65	9.8	5
3	11.2	0.28 0	0.56	1.9	0.07 5	17.0	60.0	0.99 800	3.16	0.58	9.8	6
4	7.4	0.70 0	0.00	1.9	0.07 6	11.0	34.0	0.99 780	3.51	0.56	9.4	5
...
159 4	6.2	0.60 0	0.08	2.0	0.09 0	32.0	44.0	0.99 490	3.45	0.58	10.5	5
159 5	5.9	0.55 0	0.10	2.2	0.06 2	39.0	51.0	0.99 512	3.52	0.76	11.2	6
159 6	6.3	0.51 0	0.13	2.3	0.07 6	29.0	40.0	0.99 574	3.42	0.75	11.0	6
159 7	5.9	0.64 5	0.12	2.0	0.07 5	32.0	44.0	0.99 547	3.57	0.71	10.2	5

```

159  6.0  0.31  0.47  3.6  0.06  18.0  42.0  0.99  3.39  0.66  11.0  6
8      0      7      549

```

1599 rows × 12 columns

In [3]:

```
RData.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 1599 entries, 0 to 1598
```

```
Data columns (total 12 columns):
```

#	Column	Non-Null Count	Dtype
0	fixed acidity	1599 non-null	float64
1	volatile acidity	1599 non-null	float64
2	citric acid	1599 non-null	float64
3	residual sugar	1599 non-null	float64
4	chlorides	1599 non-null	float64
5	free sulfur dioxide	1599 non-null	float64
6	total sulfur dioxide	1599 non-null	float64
7	density	1599 non-null	float64
8	pH	1599 non-null	float64
9	sulphates	1599 non-null	float64
10	alcohol	1599 non-null	float64
11	quality	1599 non-null	int64

```
dtypes: float64(11), int64(1)
```

```
memory usage: 150.0 KB
```

In [4]:

```
RX = RData.iloc[:, :-1]
```

```
Ry = RData.iloc[:, -1]
```

In [5]:

```
#RX.shape
```

```
Ry.shape
```

Out[5]:

```
(1599,)
```

Create a base model¶

In [6]:

```

def Rbaseline_model():
    model = Sequential()
    model.add(Dense(16, input_dim = 11, kernel_initializer = 'normal',
activation = 'relu'))
    model.add(Dense(1, activation = 'linear'))

```

```
model.compile(loss = 'mean_squared_error', optimizer = 'adam')
return model
```

In [7]:

```
REstimator = KerasRegressor(model = Rbaseline_model, epochs = 100, batch_size
= 5000, verbose = 0)
Rkfold = KFold(n_splits = 10)
RResults = cross_val_score(REstimator, RX, Ry, cv = Rkfold, scoring =
'neg_mean_squared_error')
print("Baseline: %.2f (%.2f)" % (RResults.mean(), RResults.std()))
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

```
WARNING:tensorflow:5 out of the last 5 calls to <function
TensorFlowTrainer.make_predict_function.<locals>.one_step_on_data_distributed
at 0x7fbbdee68280> triggered tf.function retracing. Tracing is expensive and
the excessive number of tracings could be due to (1) creating @tf.function
repeatedly in a loop, (2) passing tensors with different shapes, (3) passing
Python objects instead of tensors. For (1), please define your @tf.function
outside of the loop. For (2), @tf.function has reduce_retracing=True option
that can avoid unnecessary retracing. For (3), please refer to
https://www.tensorflow.org/guide/function#controlling_retracing and
https://www.tensorflow.org/api_docs/python/tf/function for more details.
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
```

When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
WARNING:tensorflow:6 out of the last 6 calls to <function
TensorFlowTrainer.make_predict_function.<locals>.one_step_on_data_distributed
at 0x7fbbdded70ca0> triggered tf.function retracing. Tracing is expensive and
the excessive number of tracings could be due to (1) creating @tf.function
repeatedly in a loop, (2) passing tensors with different shapes, (3) passing
Python objects instead of tensors. For (1), please define your @tf.function
outside of the loop. For (2), @tf.function has reduce_retracing=True option
that can avoid unnecessary retracing. For (3), please refer to
https://www.tensorflow.org/guide/function#controlling_retracing and
https://www.tensorflow.org/api_docs/python/tf/function for more details.
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

Baseline: -1.37 (1.03)

Improve the model by standardizing the dataset¶

In [8]:

```
REstimator = []
REstimator.append(('standardize', StandardScaler()))
REstimator.append(('mlp', KerasRegressor(model = Rbaseline_model, epochs =
100, batch_size = 5000, verbose = 0)))
pipeline = Pipeline(REstimator)
Rkfold = KFold(n_splits = 10)
RResults = cross_val_score(pipeline, RX, Ry, cv = Rkfold, scoring =
'neg_mean_squared_error')
print("Baseline: %.2f (%.2f)" % (RResults.mean(), RResults.std()))
```


Baseline: -21.86 (3.47)

Show tuning of layers and neurons (see evaluating small and larger networks)¶

Smaller Network and Wider¶

In [9]:

```
def Rbaseline_model():
    model = Sequential()
    model.add(Dense(32, input_dim = 11, kernel_initializer = 'normal',
activation = 'relu'))
    model.add(Dense(1, activation = 'linear'))

    model.compile(loss = 'mean_squared_error', optimizer = 'adam')
    return model
```

In [10]:

```
REstimator = []
REstimator.append(('standardize', StandardScaler()))
REstimator.append(('mlp', KerasRegressor(model = Rbaseline_model, epochs =
100, batch_size = 5000, verbose = 0)))
pipeline = Pipeline(REstimator)
Rkfold = KFold(n_splits = 10)
RResults = cross_val_score(pipeline, RX, Ry, cv = Rkfold, scoring =
'neg_mean_squared_error')
print("Baseline: %.2f (%.2f)" % (RResults.mean(), RResults.std()))
```

/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.

super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.

super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.

super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.

super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.

When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

Baseline: -17.52 (2.71)

Larger Network¶

In [11]:

```
def Rbaseline_model():
    model = Sequential()
    model.add(Dense(16, input_dim = 11, kernel_initializer = 'normal',
activation = 'relu'))
    model.add(Dense(8, kernel_initializer = 'normal', activation = 'relu'))
    model.add(Dense(1, activation = 'linear'))

    model.compile(loss = 'mean_squared_error', optimizer = 'adam')
    return model
```

In [12]:

```
REstimator = []
REstimator.append(('standardize', StandardScaler()))
REstimator.append(('mlp', KerasRegressor(model = Rbaseline_model, epochs =
100, batch_size = 5000, verbose = 0)))
```

```
pipeline = Pipeline(REstimator)
Rkfold = KFold(n_splits = 10)
RResults = cross_val_score(pipeline, RX, Ry, cv = Rkfold, scoring =
'neg_mean_squared_error')
print("Baseline: %.2f (%.2f)" % (RResults.mean(), RResults.std()))
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer.
When using Sequential models, prefer using an `Input(shape)` object as the
first layer in the model instead.
```

```
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

Baseline: -18.43 (4.20)

In []: