

How many training iterations are conducting in the following tuning experiment?

```
from sklearn import svm, datasets
from sklearn.model_selection import GridSearchCV
iris = datasets.load_iris()
parameters = {'kernel':('linear', 'poly', 'rbf'), 'C':[1, 5, 10]}
svc = svm.SVC() clf = GridSearchCV(svc, parameters, cv=3)
clf.fit(iris.data, iris.target)
```



Answer:

When training a model using the following code, the training stops at the end of Epoch 95. Which epoch has the best validation loss?

```
checkpoint_cb = keras.callbacks.ModelCheckpoint("my_keras_model.h5", save_best_only=True)
early_stopping_cb = keras.callbacks.EarlyStopping(patience=5, restore_best_weights=True)
history = model.fit(X_train, y_train, epochs=100, validation_data=(X_valid, y_valid), callbacks=[checkpoint_cb, early_stopping_cb])
```



Answer:

Time left 0:57:58

Question 1
Not yet answered
Marked out of 5.0
Flag question

Given the following confusion matrix, what is the prediction accuracy?

	C0	C1	C2
C0	50	5	5
C1	2	55	3
C2	6	6	48



Answer:

Next page

Rewrite the following model to build a policy network of two hidden layers (20 cells each). The network should accept 5 inputs and has 3 binary outputs.

```
model = keras.models.Sequential([  
    keras.layers.Dense(5, activation="elu", input_shape=[n_inputs]),  
    keras.layers.Dense(1, activation="sigmoid"),  
])
```



After executing the following code, the user notices that the accuracy on the train set is 95% and 50% on the test set. Which one of the following lines could work as a solution for this problem?

```
tree_clf = DecisionTreeClassifier(max_depth=5)
tree_clf.fit(X, y)
```



- tree_clf = DecisionTreeClassifier(max_depth=None)
- tree_clf = DecisionTreeClassifier(min_samples_split=2, max_depth=5)
- tree_clf = DecisionTreeClassifier(max_depth=3)
- tree_clf = DecisionTreeClassifier(max_depth=8)
- None of the other options

[Clear my choice](#)

Question **2**

Not yet answered

Marked out of 5.0

Flag question

Consider the following three layers. What is the size of the feature maps of the second convolutional layer for one input instance assuming that each number takes 4 bytes?

```
[keras.layers.Conv2D(25, 7, activation="relu", padding="same", input_shape=[28, 28, 1]),  
keras.layers.MaxPooling2D(2),  
keras.layers.Conv2D(50, 3, activation="relu", padding="same")]
```



Answer:

Next page

You need to construct a network that can predict 4 steps ahead. Rewrite the following model to achieve this objective using LSTM cells and recurrent layers of 16 cells each.

```
model = keras.models.Sequential([  
    keras.layers.SimpleRNN(20, return_sequences=True,  
        input_shape=[None, 1]),  
    keras.layers.SimpleRNN(20),  
    keras.layers.Dense(1)
```

1)



Time left 0:33:39

Question 5

Not yet answered

Marked out of 5.0

Flag question

How many trainable parameters does the following model have?

```
model = keras.models.Sequential()  
model.add(keras.layers.Dense(20, activation="relu", input_shape=[10]))  
model.add(keras.layers.Dense(100, activation="relu"))  
model.add(keras.layers.Dense(10, activation="softmax"))
```



Answer:

Next page

Rewrite the following model using SELU activation for the hidden layers and perform the necessary adjustments to get a self-normalization network.

```
model = keras.models.Sequential([
    keras.layers.Flatten(input_shape=[224, 224]),
    keras.layers.BatchNormalization(),
    keras.layers.Dense(256, activation="relu", kernel_initializer="he_normal"),
    keras.layers.BatchNormalization(),
    keras.layers.Dense(256, activation="relu", kernel_initializer="he_normal"),
    keras.layers.BatchNormalization(),
    keras.layers.Dense(1000, activation="softmax")
])
```

