```
#Import IRIS dataset from sklearn
from sklearn import datasets
# Import Random forest Logistic regression, naive bayes and knn classifier
# classes for creating stacking.
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import GaussianNB
from sklearn.neighbors import KNeighborsClassifier
#Import numpy for array based operations
import numpy as np
#Load the dataset
iris = datasets.load_iris()
#Extract data and target out of dataset
X, y = iris.data[:, 1:3], iris.target
# We will define a method to calculate accuracy of predicted output with
# known labels
def CalculateAccuracy(y_test,pred_label):
  nnz = np.shape(y_test)[0] - np.count_nonzero(pred_label - y_test)
  acc = 100*nnz/float(np.shape(y_test)[0])
  return acc
#Create a KNN classifier with 2 nearest neighbors
clf1 = KNeighborsClassifier(n_neighbors=2)
#We will create a random forest classifier with 2 decision trees
clf2 = RandomForestClassifier(n_estimators = 2,random_state=1)
#Create a Naive bayes classifier
clf3 = GaussianNB()
# Finally create a logistic regression classifier to combine prediction from
# above classifiers.
lr = LogisticRegression()
#Now we will Train all first level classifiers
clf1.fit(X, y)
clf2.fit(X, y)
clf3.fit(X, y)
#Predict the labels for input data by all the classifier; print their
# accuracy and store the prediction into an array (f1,f2,f3)
f1 = clf1.predict(X)
acc1 = CalculateAccuracy(y, f1)
print("accuracy from KNN: "+str(acc1) )
f2 = clf2.predict(X)
acc2 = CalculateAccuracy(y, f2)
print("accuracy from Random Forest: "+str(acc2) )
f3 = clf3.predict(X)
acc3 = CalculateAccuracy(y, f3)
print("accuracy from Naive Bayes: "+str(acc3) )
# Combine the predictions into a single array and transpose the array to
# match input shape of or classifier.
f = [f1,f2,f3]
f = np.transpose(f)
# Now train the classifier
lr.fit(f, y)
final = lr.predict(f)
# Calculate and print the accuracy of final classifier
acc4 = CalculateAccuracy(y, final)
print("accuracy from Stacking: "+str(acc4) )
```

```
accuracy from KNN: 96.66666666666667
accuracy from Random Forest: 94.66666666666667
accuracy from Naive Bayes: 92.0
accuracy from Stacking: 97.33333333333333
```