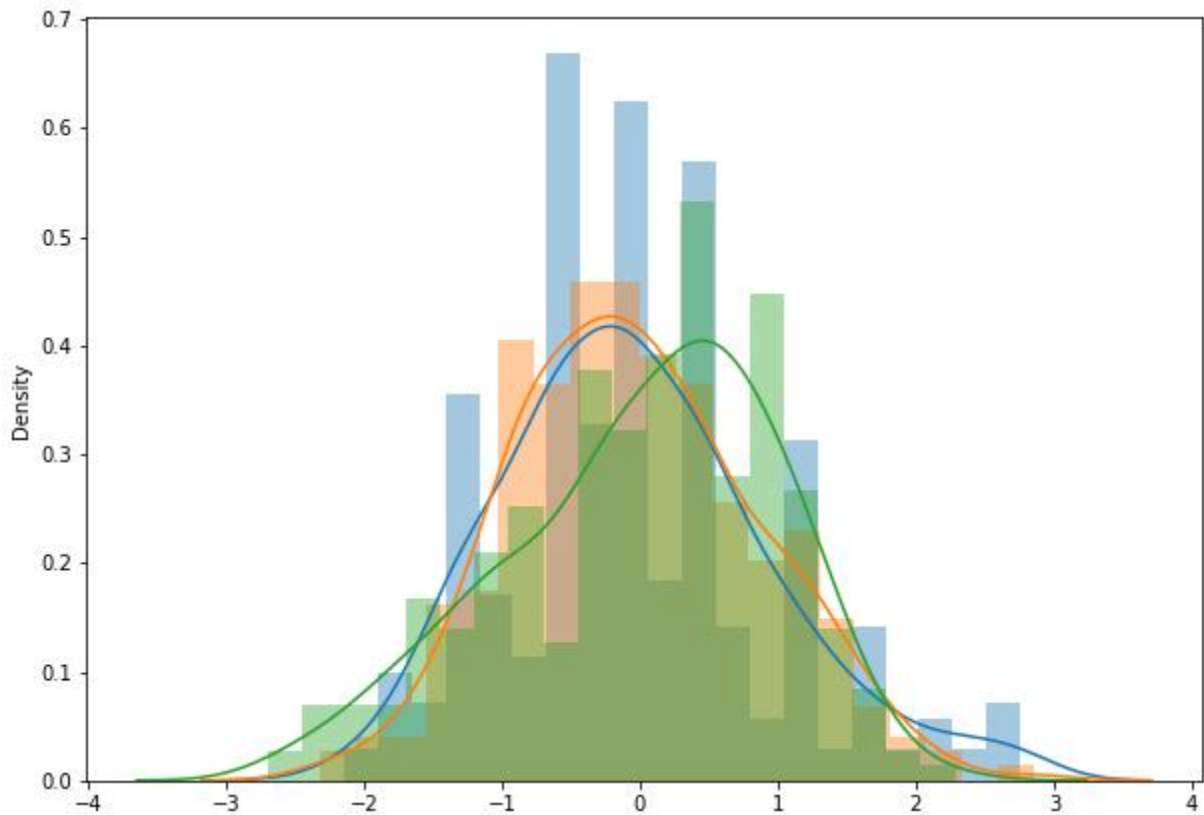


جواب سوال 5 و 6 و 7

```
# perform Gaussian naive bayes with package
x1 = ds[['trestbps','chol','thalach']]
X_train, X_test, y_train, y_test = tts( x1 , y , test_size = 0.2, random_state = 0)
print('X_train.shape =', X_train.shape)
print('X_test.shape =', X_test.shape)
print('y_train.shape =', y_train.shape)
print('y_test.shape =', y_test.shape)
plt.figure(figsize=(10,7))
for i in x1.columns:
    sns.distplot(x1[i],kde_kws={"label": i}, bins=20)
gnb = GaussianNB()
gnb.fit(X_train , y_train)
```

```
X_train.shape = (229, 3)
X_test.shape = (58, 3)
y_train.shape = (229,)
y_test.shape = (58,)
```



```
print("Recall : ",gnb.score(X_test,y_test))
```

Recall : 0.7241379310344828

نمودار مدل را رسم کردیم و با استفاده از پکیج **GaussianNB** توانستیم مدل را روی داده های آموزشی پیاده سازی کنیم و در خروجی مقدار خطای مدل را به دست آوردیم که برابر است با 0.724 .

```
pred = gnb.predict(X_test)
print("F1-score : ",f1_score(y_test, pred, average='macro'))
print("Precision : ",precision_score(y_test, pred, average='macro'))
print("Recall : ",recall_score(y_test, pred, average='macro'))
```

```
F1-score : 0.7187878787878788
Precision : 0.7425997425997426
Recall : 0.7241379310344827
```

اکنون پکیج های (**f1\_score** و **precision\_score** و **recall\_score**) را روی مدل گوسی پیاده سازی می کنیم و خروجی را ملاحظه میکنیم.

حال می خواهیم مدل دیگری را برای داده ها امتحان کنیم. برای مثال : مدل برنولی  
با استفاده از پکیج **BernoulliNB** مدل را روی داده های آموزش پیاده سازی می کنیم:

```
# perform Bernoulli naive bayes with package
x1 = ds[['trestbps','chol','thalach']]
X_train, X_test, y_train, y_test = tts( x1 , y , test_size = 0.2, random_state = 0)
print('X_train.shape =', X_train.shape)
print('X_test.shape =', X_test.shape)
print('y_train.shape =', y_train.shape)
print('y_test.shape =', y_test.shape)
bnb = BernoulliNB()
bnb.fit(X_train , y_train)

X_train.shape = (229, 3)
X_test.shape = (58, 3)
y_train.shape = (229,)
y_test.shape = (58,)
BernoulliNB(alpha=1.0, binarize=0.0, class_prior=None, fit_prior=True)
```

```
# Calculate the error value of the model Bernoulli
pred = bnb.predict(X_test)
print("F1-score : ",f1_score(y_test, pred, average='macro'))
print("Precision : ",precision_score(y_test, pred, average='macro'))
print("Recall : ",recall_score(y_test, pred, average='macro'))
```

```
F1-score : 0.8102884329467737
Precision : 0.8107142857142857
Recall : 0.8103448275862069
```

مقدار خطای به دست آمده در این مدل 0.810 خواهد بود که با توجه به این که مقدار خطا نسبت به مدل گوسی افزایش پیدا کرده است پس مدل مناسبی نیست.