BMF

AFP-ANN

def build\_model():

model = Sequential()

model.add(layers.Dense(400, input\_dim=512))

model.add(Dropout(0.3))

model.add(layers.Dense(200))

model.add(Dropout(0.3))

model.add(layers.Dense(100))

model.add(Dropout(0.3))

model.add(layers.Dense(1))

sgd = keras.optimizers.SGD(lr=0.0001)

model.compile(loss='mse',optimizer='sgd', metrics=['RootMeanSquaredError'])

return model

model=KerasRegressor(build\_fn=build\_model,epochs=150,batch\_size=256,verbose=0)

model.fit(X\_train,y\_train,verbose=0)

ECFP-ANN

def build\_model():

model = Sequential()

model.add(layers.Dense(300, input\_dim=1024))

model.add(Dropout(0.3))

model.add(layers.Dense(150))

model.add(Dropout(0.3))

model.add(layers.Dense(75))

model.add(Dropout(0.3))

model.add(layers.Dense(1))

sgd = keras.optimizers.SGD(lr=0.0001)

model.compile(loss='mse',optimizer='sgd', metrics=['RootMeanSquaredError'])

return model

model=KerasRegressor(build\_fn=build\_model,epochs=150,batch\_size=256,verbose=0)

model.fit(X\_train,y\_train,verbose=0)

MACCS-ANN

def build\_model():

model = Sequential()

model.add(layers.Dense(300, input\_dim=167))

model.add(Dropout(0.05))

model.add(layers.Dense(150))

model.add(Dropout(0.05))

model.add(layers.Dense(75))

model.add(Dropout(0.05))

model.add(layers.Dense(1))

sgd = keras.optimizers.SGD(lr=0.001)

model.compile(loss='mse',optimizer='sgd', metrics=['RootMeanSquaredError'])

return model

model=KerasRegressor(build\_fn=build\_model,epochs=150,batch\_size=256,verbose=0)

model.fit(X\_train,y\_train,verbose=0)

Modred-ANN

def build\_model():

model = Sequential()

model.add(layers.Dense(200, input\_dim=1277))

model.add(Dropout(0.2))

model.add(layers.Dense(100))

model.add(Dropout(0.2))

model.add(layers.Dense(50))

model.add(Dropout(0.2))

model.add(layers.Dense(1))

sgd = keras.optimizers.SGD(lr=0.001)

model.compile(loss='mse',optimizer='sgd', metrics=['RootMeanSquaredError'])

return model

model=KerasRegressor(build\_fn=build\_model,epochs=150,batch\_size=256,verbose=0)

model.fit(X\_train,y\_train,verbose=0)

RDKFP-ANN

def build\_model():

model = Sequential()

model.add(layers.Dense(500, input\_dim=1024))

model.add(Dropout(0.2))

model.add(layers.Dense(250))

model.add(Dropout(0.2))

model.add(layers.Dense(175))

model.add(Dropout(0.2))

model.add(layers.Dense(1))

sgd = keras.optimizers.SGD(lr=0.0001)

model.compile(loss='mse',optimizer='sgd', metrics=['RootMeanSquaredError'])

return model

model=KerasRegressor(build\_fn=build\_model,epochs=150,batch\_size=256,verbose=0)

model.fit(X\_train,y\_train,verbose=0)

AFP-GBR

fit1=GradientBoostingRegressor(learning\_rate=0.0489,max\_depth=6, max\_features='log2',min\_samples\_leaf=1, min\_samples\_split=2,n\_estimators=350,random\_state=42)

ECFP-GBR

fit1=GradientBoostingRegressor(learning\_rate=0.0186,max\_depth=10, max\_features='sqrt',min\_samples\_leaf= 1,min\_samples\_split=3,n\_estimators=750,random\_state=42)

MACCS-GBR

fit1=GradientBoostingRegressor(learning\_rate=0.0192,max\_depth=10, max\_features='sqrt',min\_samples\_leaf= 1,min\_samples\_split=2,n\_estimators=950,random\_state=42)

Modred-GBR

fit1=GradientBoostingRegressor(learning\_rate=0.1,max\_depth=2, max\_features='sqrt',min\_samples\_leaf= 5,min\_samples\_split=5,n\_estimators=50,random\_state=42)

RDKFP-GBR

fit1=GradientBoostingRegressor(learning\_rate=0.1969,max\_depth=3, max\_features='sqrt',min\_samples\_leaf= 20,min\_samples\_split=2,n\_estimators=1000,random\_state=42)

AFP-RF

fit1=RandomForestRegressor(max\_depth=10, max\_features='sqrt',min\_samples\_leaf= 1,min\_samples\_split=2,n\_estimators=100,random\_state=42)

ECFP-RF

fit1=RandomForestRegressor(max\_depth=10, max\_features='sqrt',min\_samples\_leaf= 1,min\_samples\_split=2,n\_estimators=500,random\_state=42)

MACCS-RF

fit1=RandomForestRegressor(max\_depth=7, max\_features='log2',min\_samples\_leaf= 1,min\_samples\_split=2,n\_estimators=100,random\_state=42)

Mordred-RF

fit1=RandomForestRegressor(max\_depth=8, max\_features='sqrt',min\_samples\_leaf= 5,min\_samples\_split=4,n\_estimators=300,random\_state=42)

RDKFP-RF

fit1=RandomForestRegressor(max\_depth=10, max\_features='log2',min\_samples\_leaf= 1,min\_samples\_split=2,n\_estimators=100,random\_state=42)

AFP-SVM

fit1=SVR(C=1000 , kernel='poly',max\_iter=10000,coef0=0,degree=2,gamma=0.0042)

ECFP-SVM

fit1=SVR(C=0.0134, kernel='poly',max\_iter= 7500,coef0=-1,degree=2,gamma=6.3564)

MACCS-SVM

fit1=SVR(C=1000, kernel='poly',max\_iter= 10000,coef0=0,degree=2,gamma=0.0036)

Mordred-SVM

fit1=SVR(C=588.5231, kernel='rbf',max\_iter= 4300,gamma=0.0003)

RDKFP-SVM

fit1=SVR(C=0.1, kernel='poly',max\_iter= 1000,coef0=0,degree=1,gamma=0.03)