```
import numpy as np
import pandas as pd
import sklearn
from sklearn import svm
from sklearn import metrics
from sklearn.model_selection import train_test_split
#data url load the data in
data_url = "https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.data"
df = pd.read_csv(data_url, delimiter = ",", header = None)
display(df)
y = df[1].values
x = df.iloc[:, 2:32]
x = np.array(x)
rs_cols = ["train accuracy", "test accuracy", "precision", "recall"] #column names for the result table
rs_rows = ["SVM1", "SVM2", "SVM3"] #row names for the result table
rs_table = pd.DataFrame(columns=rs_cols, index=rs_rows) #creating the result table
display(rs_table)
print("Wait a moment computing SVM1 Values..")
clf = svm.SVC(C = 1e10, kernel = 'linear') # Linear Kernel
train_acc = []
test_acc = []
prec = []
recall = []
for i in range(20):
    (x_train, x_test, y_train, y_test) = train_test_split(x, y, test_size = 0.3)
    scaler = sklearn.preprocessing.StandardScaler()
    x train = scaler.fit transform(x train)
    x_test = scaler.fit_transform(x_test)
    clf = clf.fit(x_train, y_train)
    #for train dataset
    y_train_pred = clf.predict(x_train)
    train_acc.append(metrics.accuracy_score(y_train, y_train_pred))
    #for test dataset
    y_test_pred = clf.predict(x_test)
    test_acc.append(metrics.accuracy_score(y_test, y_test_pred))
    prec.append(metrics.precision_score(y_test, y_test_pred, pos_label = 'M'))
    recall.append(metrics.recall_score(y_test, y_test_pred, pos_label = 'M'))
#compute average of 20 performance
mean_train_acc = np.mean(train_acc)
mean_test_acc = np.mean(test_acc)
mean_prec = np.mean(prec)
mean_recall = np.mean(recall)
rs_table.loc['SVM1']['train accuracy'] = mean_train_acc
rs_table.loc['SVM1']['test accuracy'] = mean_test_acc
rs_table.loc['SVM1']['precision'] = mean_prec
rs_table.loc['SVM1']['recall'] = mean_recall
print(rs_table)
print("Wait a Moment Calculating for SVM2....")
clf = svm.SVC(C = 1e10, kernel = 'rbf') # RBF Kernel
train_acc = []
test_acc = []
prec = []
```

```
recall = []
for i in range(20):
    (x_train, x_test, y_train, y_test) = train_test_split(x, y, test_size = 0.3)
    scaler = sklearn.preprocessing.StandardScaler()
    x_train = scaler.fit_transform(x_train)
    x_test = scaler.fit_transform(x_test)
    clf = clf.fit(x_train, y_train)
    #for train dataset
    y_train_pred = clf.predict(x_train)
    train_acc.append(metrics.accuracy_score(y_train, y_train_pred))
    #for test dataset
    y_test_pred = clf.predict(x_test)
    test_acc.append(metrics.accuracy_score(y_test, y_test_pred))
    prec.append(metrics.precision_score(y_test, y_test_pred, pos_label = 'M'))
    recall.append(metrics.recall_score(y_test, y_test_pred, pos_label = 'M'))
#compute average of 20 performance
mean_train_acc = np.mean(train_acc)
mean_test_acc = np.mean(test_acc)
mean_prec = np.mean(prec)
mean_recall = np.mean(recall)
rs_table.loc['SVM2']['train accuracy'] = mean_train_acc
rs_table.loc['SVM2']['test accuracy'] = mean_test_acc
rs_table.loc['SVM2']['precision'] = mean_prec
rs_table.loc['SVM2']['recall'] = mean_recall
print(rs_table)
print("Calculating for SVM3....")
clf = svm.SVC(C = 10, kernel = 'rbf') # RBF Kernel with varying C value
train_acc = []
test_acc = []
prec = []
recall = []
for i in range(20):
    (x_train, x_test, y_train, y_test) = train_test_split(x, y, test_size = 0.3)
    scaler = sklearn.preprocessing.StandardScaler()
    x_train = scaler.fit_transform(x_train)
    x_test = scaler.fit_transform(x_test)
    clf = clf.fit(x_train, y_train)
    #for train dataset
    y_train_pred = clf.predict(x_train)
    train_acc.append(metrics.accuracy_score(y_train, y_train_pred))
    #for test dataset
    y_{test_pred} = clf.predict(x_{test})
    test_acc.append(metrics.accuracy_score(y_test, y_test_pred))
    \verb|prec.append(metrics.precision_score(y_test, y_test_pred, pos_label = 'M'))| \\
    recall.append(metrics.recall_score(y_test, y_test_pred, pos_label = 'M'))
#compute average of 20 performance
mean_train_acc = np.mean(train_acc)
mean_test_acc = np.mean(test_acc)
mean_prec = np.mean(prec)
mean_recall = np.mean(recall)
rs_table.loc['SVM3']['train accuracy'] = mean_train_acc
rs_table.loc['SVM3']['test accuracy'] = mean_test_acc
rs_table.loc['SVM3']['precision'] = mean_prec
rs_table.loc['SVM3']['recall'] = mean_recall
```

print(rs_table)

		_	_	_	_	_	_	_	_	_								
	0	1	2	3	4	5	6	7	8	9	•••	22	23	24	25	26	27	
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710		25.380	17.33	184.60	2019.0	0.16220	0.66560	0.
1	842517	М	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017		24.990	23.41	158.80	1956.0	0.12380	0.18660	0.2
2	84300903	М	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790		23.570	25.53	152.50	1709.0	0.14440	0.42450	0.4
3	84348301	М	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520		14.910	26.50	98.87	567.7	0.20980	0.86630	0.6
4	84358402	М	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.19800	0.10430		22.540	16.67	152.20	1575.0	0.13740	0.20500	0.
					•••		•••		•••			•••				•••	•••	
564	926424	M	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890		25.450	26.40	166.10	2027.0	0.14100	0.21130	0.4
565	926682	M	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791		23.690	38.25	155.00	1731.0	0.11660	0.19220	0.
566	926954	М	16.60	28.08	108.30	858.1	0.08455	0.10230	0.09251	0.05302		18.980	34.12	126.70	1124.0	0.11390	0.30940	0.
567	927241	М	20.60	29.33	140.10	1265.0	0.11780	0.27700	0.35140	0.15200		25.740	39.42	184.60	1821.0	0.16500	0.86810	0.
					47.00	404.0	0.05262	0.04362	0.00000	0.00000		9.456	30.37	59.16	268 6	0.08996	0.06444	Ω
568 569 re	ows × 32 col	umn			47.92			_	0.00000	0.00000	•••		00.01	00.10	200.0			0.
	ows × 32 col	umn:	S			precisi	on recal	1 11	0.00000	0.0000				00.10	20010			0.
569 r	ows × 32 col train a	umn:	s racy t		curacy	precisi Na	on recal	1 11 N	0.0000	0.0000				66.16	200.0			0.
569 r	ows × 32 col train a //1	umn:	s r acy t NaN		curacy NaN	precisi Na Na	on recal	1 II.	0.0000	0.0000				66.16	200.0			0.
SVM SVM	ows × 32 columnation and the state of the st	umn:	s racy t NaN NaN NaN uting S	est acc	NaN NaN NaN NaN	precisi Na Na Na	on recal aN Na aN Na aN Na	1 II.	0.0000	0.0000				66.16	200.0			0.
SVN SVN SVN Wait	train a 11 12 13 a moment of train according accordi	umn: iccui	s racy t NaN NaN NaN uting !	eest acc SVM1 Va accura	NaN NaN NaN NaN olues	precisi Na Na Na ision	on recal aN Na aN Na aN Na recall	1 II.	0.0000	0.0000				66.16	200.0			0.
SVN SVN SVN Wait	train a 11 12 13 a moment of train according to the contrain according to the contract of	compurac	nacy t NaN NaN NaN uting s y test	SVM1 Va accura 0.9473	NaN NaN NaN NaN olues acy prec	precisi Na Na Na ision 11634 (on recal Na NA NA NA Pecall 9.951289	1 II.	0.0000	0.0000				00.10	2000			0.
SVN SVN SVN Wait	ows × 32 columns and 1 11 12 13 13 14 15 16 17 17 18 18 18 18 18 18 18 18	umn: iccui	nacy t NaN NaN NaN uting s y test 0	SVM1 Va accura 0.9473	NaN NaN NaN NaN olues	precisi Na Na Na ision	on recal aN Na aN Na aN Na recall	1 II.	0.0000	0.0000				00.10				0.
SVN SVN SVN Wait SVM1 SVM2 SVM3	ows × 32 columns and 1 11 12 13 13 14 15 16 17 17 18 18 18 18 18 18 18 18	comp compurac 1. Na	s racy t NaN NaN NaN uting ! y test 0 N	SVM1 Va accura 0.9473 N	NaN NaN NaN NaN Nlues icy prec 68 0.9 laN	precisi Na Na Na ision 11634 (NaN	on recal Na	1 II.	0.0000	0.0000				00.10				0.
SVN SVN SVN Wait SVM1 SVM2 SVM3 Wait	ows × 32 columns and 11 12 13 13 14 15 16 17 17 18 18 18 18 18 18 18 18	compurac 1. Na Na Calc	s racy t NaN NaN uting 9 y test 0 N N ulating y test	SVM1 Va accura 0.9473 N N g for S accura	NaN NaN NaN Nalues cy prec 68 0.9 laN laN NMA NAN NAN NAN NAN NAN NAN NAN NAN NA	precisi Na Na Na ision 11634 NaN NaN NaN Nision	on recal aN Na aN Na aN Na recall 0.951289 NaN NaN recall	1 II.	0.0000	0.0000				00.10				0.
SVN SVN SVN Wait SVM1 SVM2 SVM3 Wait	www x 32 columnation at the state of the sta	compurace 1. Na Na Calcurace 1. 1.	s racy t NaN NaN NaN uting ! y test 0 N ulating y test	SVM1 Va accura 0.9473 N N g for S accura 0.9473	NaN NaN NaN Nlues locy prec loa NaN NaN Nues Nues Nues Nues Nues Nues Nues Nue	precisi Na Na Na ision 11634 NaN NaN ision 11634 6	on recal aN Na aN Na recall 9.951289 NaN NaN recall 9.951289	1 II.	0.0000	0.0000				00.10				0.
SVN SVN Wait SVM1 SVM2 SVM3 Wait SVM1 SVM1 SVM2	ows × 32 columns and 1 11 12 13 13 14 15 16 17 17 18 18 18 18 18 18 18 18	compurace 1. Na Na Calc	racy t NaN NaN uting ! y test 0 N uulatini y test 0 0 0	SVM1 Va accura 0.9473 N N g for S accura 0.9473 0.9532	NaN NaN NaN lues cy prec 168 0.9 1aN NaN NaN NaN NaN NaN NaN NaN NaN NaN	precisi Na Na Na ision 11634 (NaN NaN ision 11634 (25467 (on recal aN Na aN Na recall 0.951289 NaN NaN recall 0.951289	1 II.	0.0000	0.0000				00.10				
SVN SVN Wait SVM1 SVM2 SVM3 Wait SVM1 SVM2 SVM3 SVM2 SVM3	ows × 32 col train a M1 M2 M3 a moment of train accol train accol train accol train accol	compurace 1. Na Na Calc urac 1. Na	s racy t NaN NaN uting 9 y test 0 N N ulating ulating 0 0	SVM1 Va accura 0.9473 N N g for S accura 0.9473 0.9532	NaN NaN NaN Nlues locy prec loa NaN NaN Nues Nues Nues Nues Nues Nues Nues Nue	precisi Na Na Na ision 11634 NaN NaN ision 11634 6	on recal aN Na aN Na recall 9.951289 NaN NaN recall 9.951289	1 II.	0.0000	0.0000				Control				
SVN SVN Wait SVM1 SVM2 SVM3 Wait SVM1 SVM2 SVM3 SVM2 SVM3	ows × 32 col train a 11 12 13 a moment of train accol train accol train accol ulating for	compurace 1. Na Calcurace 1. Na Salcurace 2. Salcurace 3. Salcurace 5. Salcurace 5. Salcurace 6. Salcurace 7. Salcurace 8.	s racy t NaN NaN NaN uting 9 y test 0 N N ulating y test 0 0 N M M M M M M M M M M M M M M M M M	SVM1 Va accura 0.9473 N g for S accura 0.9473 0.9532 N	NaN NaN NaN Nalues cy prec 68 0.9 laN laN NaN NM2 cy prec 68 0.9 laN laN NM2	precisi Na Na Na ision 11634 (NaN NaN ision 11634 (25467 (NaN	on recal aN Na aN Na recall 0.951289 NaN NaN recall 0.951289	1 II.	0.0000	0.0000				00.10				
SVN SVN Wait SVM1 SVM2 SVM3 Wait SVM1 SVM2 SVM3 SVM2 SVM3	ows × 32 color train a 11 12 13 a moment of train according to the color train according train according train according train according train according trai	compurace 1. Na Calcurace 1. Na Salcurace 2. Salcurace 3. Salcurace 5. Salcurace 5. Salcurace 6. Salcurace 7. Salcurace 8.	s racy t NaN NaN uting 9 y test 0 N ulating y test 0 0 N ulating y test	SVM1 Va accura 0.9473 N g for S accura 0.9473 0.9532 N	NaN NaN NaN Nalues cy prec 68 0.9 laN laN NaN NM2 cy prec 68 0.9 laN laN NM2	precisi Na Na Na ision 11634 6 NaN NaN ision 11634 225467 6 NaN ision	on recal aN Na aN Na recall 0.951289 NaN NaN recall 0.951289 NaN NaN	1 II.	0.0000	0.0000				00.10				
SVNW SVNW Wait SVM1 SVM2 SVM3 Wait SVM1 SVM1 SVM1 SVM1 SVM1 SVM1 SVM2 SVM3 Calculations	train a 11 12 13 a moment of train accountain accou	compurace 1. Na Calcurace 1. Na Salcurace 2. Salcurace 3. Salcurace 4. Salcurace 5. Salcurace 6. Salcurace 7. Salcurace 8. Salcurace 8. Salcurace 9.	racy t NaN NaN NaN uting 9 y test 0 N N ulating y test 0 0 N ulating y test 0 0 y test	SVM1 Va accura 0.9473 N N g for S accura 0.9473 0.9532 N accura	NaN NaN NaN Nalues cy prec 68 0.9 laN laN locy prec 68 0.9 16 0.9 laN cy prec	precisi Na Na Na ision 11634 (NaN NaN ision 11634 (25467 (NaN ision 11634 (on recal aN Na aN Na aN Na recall 0.951289 NaN NaN recall 0.951289 0.950779 NaN recall	1 II.	0.0000	0.0000				00.10				0.

Start coding or generate with AI.