

23 2022

лф ь млл н йй бв vбмиvкш
. + +. +3 Л кл ш в з m п вккл л вкпвиивз п

млб лпл з в : 11+0+2 вб л вфв з лвл н дл квв б пйь млл виьй в
: пвй пвз , к лнй пвз
4 :
9
2 72
9

16 2022

10

2022

.+ в плбб vбмивкш пнпв пнв Л

,

+ и кбнпвй шв нвдпшц пш л пфкбъ млбб vбмивкв

3-6.	Дк пш7 Мй вш7 и бвш7 -
3-8.	Дк пшц Мй вшц и бвшц

--	--

0+Lлбвнг квббб vбмивкш

1. плнвь б зт п вккл л бкпвиивзп

2. и бкклв л пфкбв Л +

3. влбш зи б бз vbb б ккшт

(kNN)

4. Знв ь ли б лз OL ros b +

5. К б кшж жвл збж зи б бз пли

6. Ил б пфв з ь нв нв бь

7. Д б ф нв нв bb+

8. влбш зи пвнвд vbb

EM-

(k-means).

Fuzzy c-means

9. Квжл вив ь м н бб й б зт п вкклй бкпвиивзпв

10. ливвблкшв б вкпвфв з вв и лнвпйш

11. ви вкбв мл и вй й влб й в млв з

12. **вплбш мнвб п ивкбъ в л н лпз в квфвз лж вк лнй vbb бкпвиивз пт ицшт в пвй т**

13. **кпвиивз пт ицштв йкл л вкпкшв в пвйш L +**

: Java Agent Development Framework (Jade), ABLE, REPAST.

14. **нвй вквкбъ L л н лпз в вди н г вкбж в з лймщпвнклий днкбб**

4+Пвй пвфв з вжми к

1		4	1		1	2
2		4	1		1	2
3		4	1		1	2
4	curve).	4	1		1	2
5		6	2		2	2
6		4	1		1	2
7		4	1		1	2
8		6	2	-	2	2
9		6	1	-	1	4
10		6	1	-	1	4
11		6	1	-	1	4
12		6	1	-	1	4
13		6	1	-	1	4
14		6	1	-	1	4

	ПЛ Л	72	16		16	13+27

5+ ббш л н дп пвицкжбвьпвицкпб

Д кьпвь ивз вблккл лпбм

1. пльвь б зт п вккл л бкпвиивзп . и бкклв л пфкбв Л +

2. влбш зи б бз вбб б ккшт+З нв ь ли в лз (ROC curve).

3. К б кшж жвл збж зи б бз пн+

4. Ил в пфв з ь нв нв бь+Д б ф нв нв бб+

5. влбш зи пвнд вбб+

EM-

(k-means).

6. Квжнл вив ь м н бв й б зт п вкклй бкпвиивзпв+ ливвблкшв б вквпфв з вв и лнрпйш+

7. ви вкбв мнл ивй йвлб йв млб з + влбш мнб п ивкбв б л н лзв квфвиз.лж бк лнй вбб бкпвиивзптицкшт б пвй т+

8. $L + \text{нйвквкь } L$ л н лз в
вдл н г вквж в з лй мщпвнклей днвкбб+

Д кьпъ вйвк н з л лпвм

н з прфв з ввд кьпъ

И лн плнкшвн лш

ROC
3
4
5
7
8

И . Л н лз в тлбкшт б ккшт бий й и вккл л л пфкь. влбш зи в вз vbb
б ккшт +

- dataset Titanic: test.csv train.csv
(<https://russianblogs.com/article/49401398867/>).

Д б кбв 1
test.csv train.csv

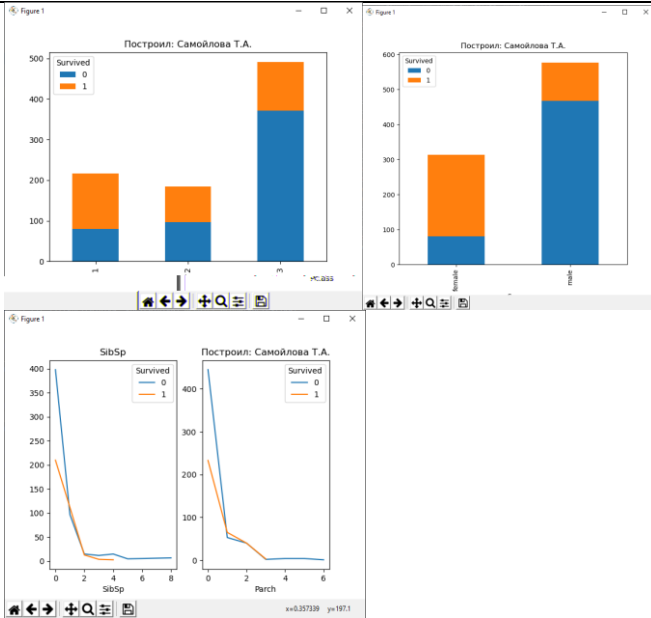
•

-
-
-

:

	Survived	Pclass	Sex	SibSp	Parch	Fare	Embarked
0	0	3	male	1	0	7.2500	S
1	1	1	female	1	0	71.2833	C
2	1	3	female	0	0	7.9250	S
3	1	1	female	1	0	53.1000	S
4	0	3	male	0	0	8.0500	S

	Survived	Pclass	Sex	SibSp	Parch	Fare	Embarked
0	0	3	1	1	0	7.2500	2
1	1	1	0	1	0	71.2833	0
2	1	3	0	0	0	7.9250	2
3	1	1	0	1	0	53.1000	2
4	0	3	1	0	0	8.0500	2



Д б квв 2

python-

age,salary,house,class

- 18,25,1,1
- 22,100,1,1
- 30,10,0,0
- 32,120,0,1
- 24,15,1,0
- 25,22,1,1
- 32,20,0,0
- 19,15,1,0
- 52,135,0,1

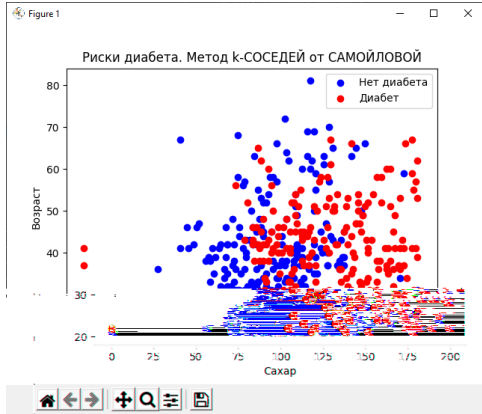
.....

Д б квв3 .

Д б квв4

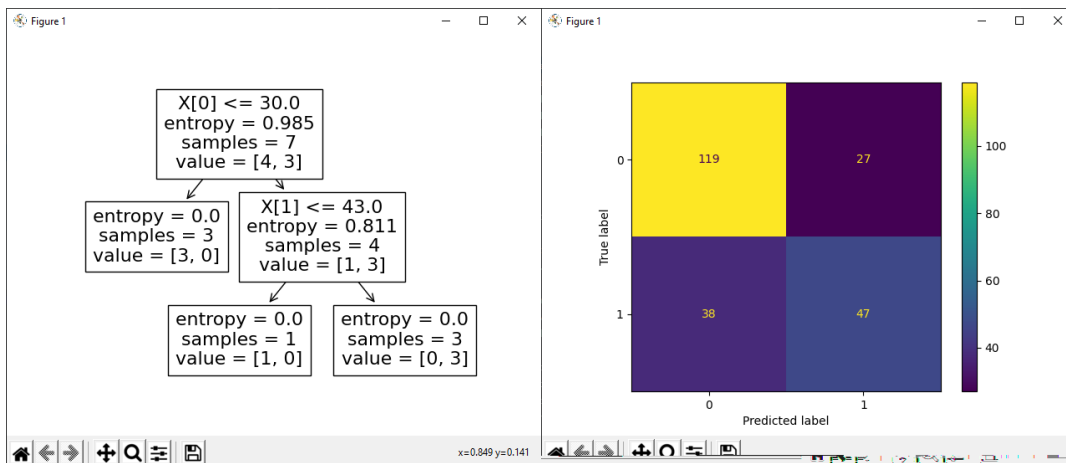
F-

Б ккшвбиль 3 мвнйвкпл : UCI Machine Learning Repository <http://archive.ics.uci.edu/ml/>



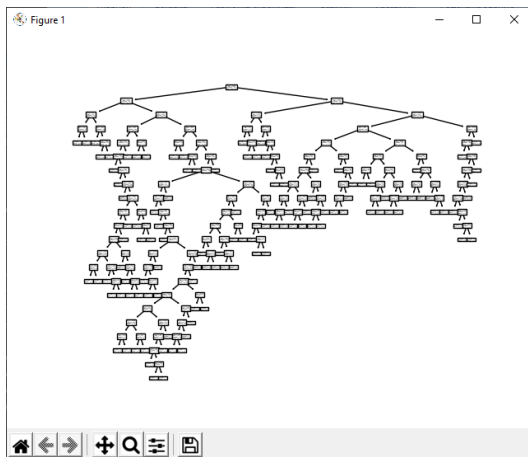
Д б квв5

ID



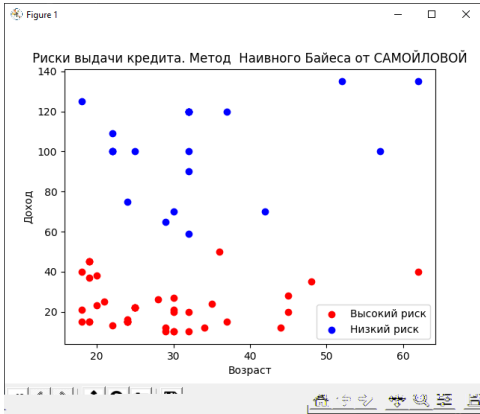
Д б квв6

C CART.

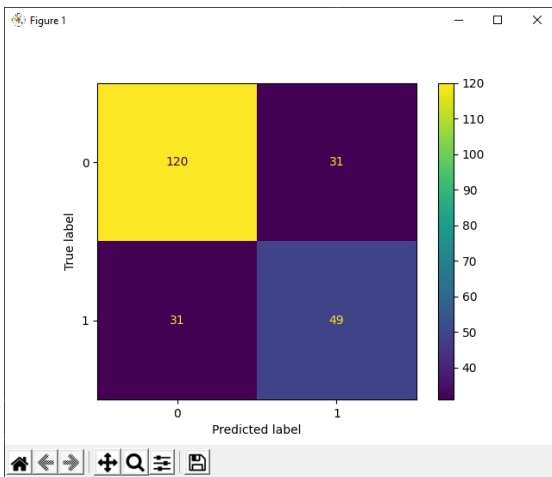


И 2 и бкклв л пфкбв йвлблй к б кл л жв . Лувкз з фв п й и бккл л л пфкбв йвлблй ROC к ивд . Дб квв .

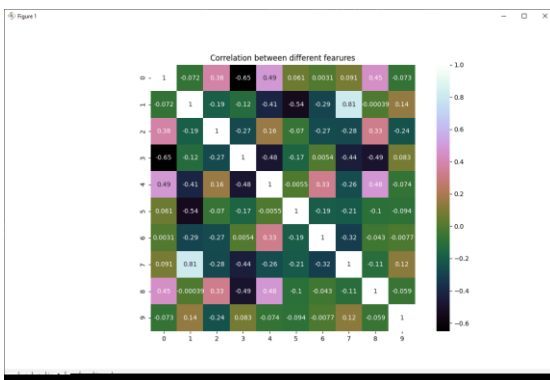
f-

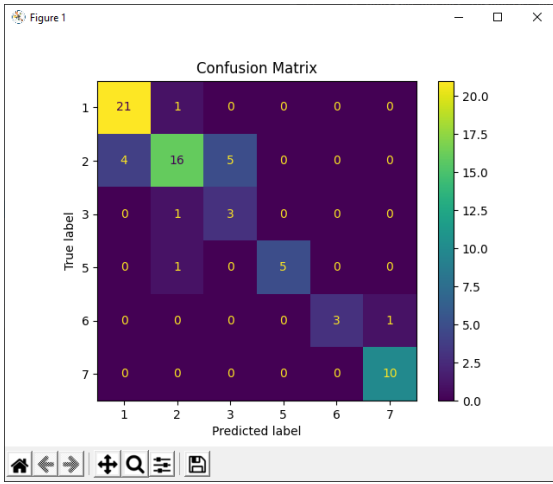


Дб квв



Дб квв0+

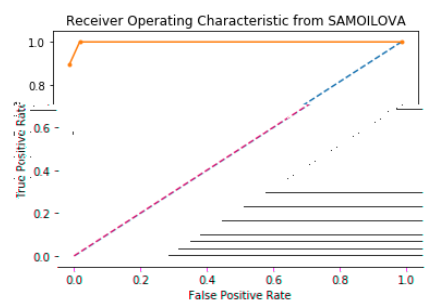
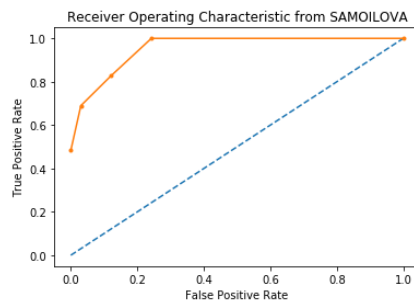
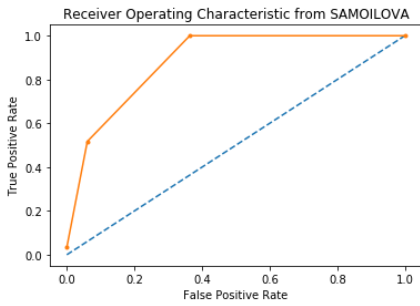




Д б квв4

f-

ROC-



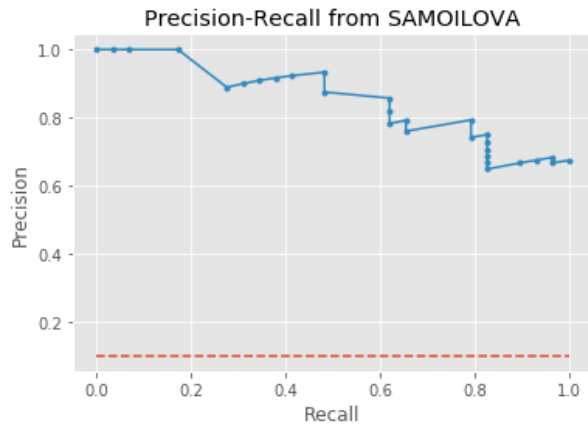
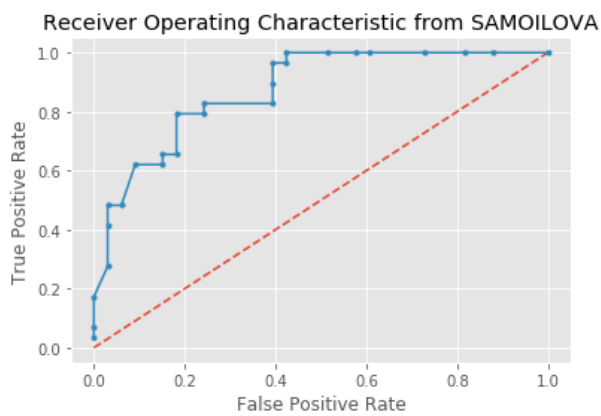
Д б квв5

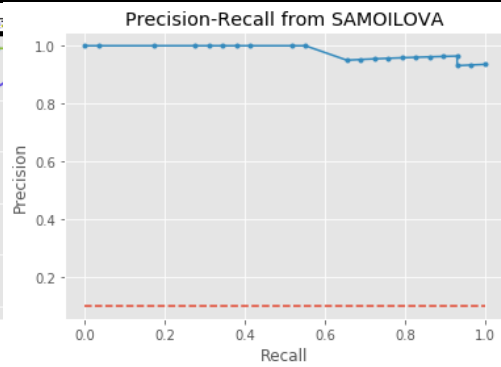
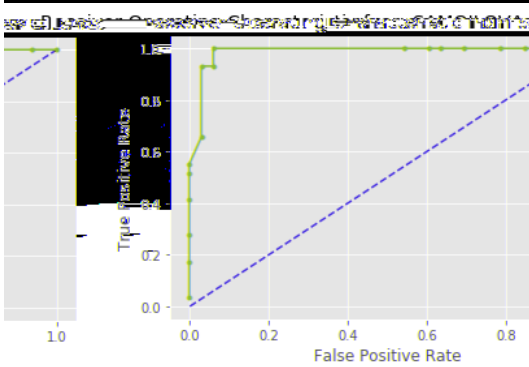
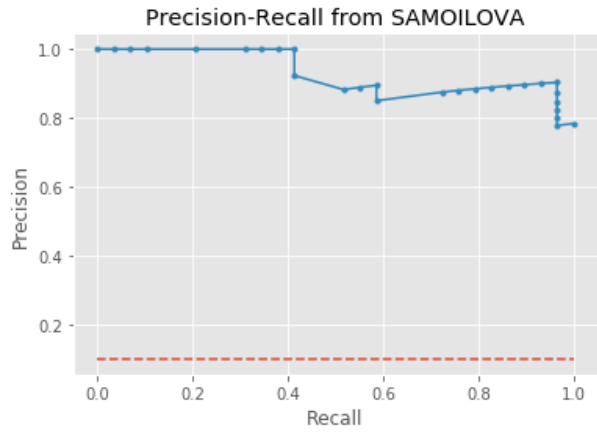
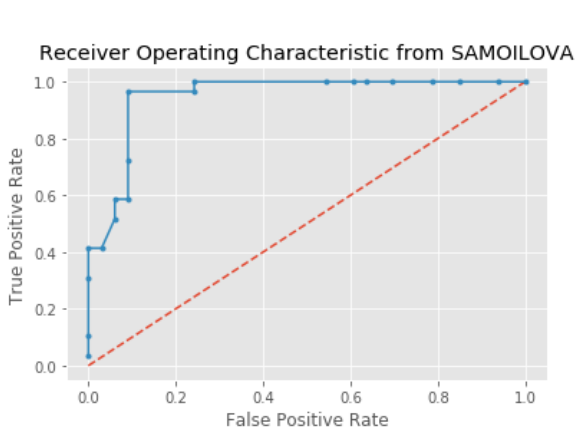
ROC -

PR-

бий н джшт дж фвквж

ит вкш бвнв :





Д б квв б.

AdaBoostClassifier
CSV

f- ROC AUC.

```

18 #names = f'age', 'sex'
19 dataframe = pd.read_csv('data.csv')
20
21 print('Введены параметры')
22 #обучаем модель
23 print('Формат обучения')
24 print(dataframe.head())
25 array = dataframe.values
26 X = array[:,0:3]
27 Y = array[:,3]
28 x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.3, random_state=0)
29 #создаем экземпляр
30 model = AdaBoostClassifier()
31 #обучаем модель
32 model.fit(x_train, y_train)
33 #предсказываем вероятности
34 probs = model.predict_proba(x_test)
35 print(model.predict(x_test))
36 print(probs)
37 # keep probabilities from the raw predicted results
38 probs = probs[:,1]
39 # calculate AUC
40 my_auc = roc_auc_score(y_test, probs)
41 print('AUC: %.3f' % my_auc)
42 # calculate roc curve
43 fpr, tpr, thresholds = roc_curve(y_test, probs)
44 # plot no skill
45 plt.plot([0, 1], [0, 1], 'r--')
46 # plot the roc curve for the model
47 plt.plot(fpr, tpr, 'b-')
48 plt.title('Receiver Operating Characteristic from SAMOILOVA')
49 plt.ylabel('True Positive Rate')
50 plt.xlabel('False Positive Rate')
51 plt.show()
52 # calculate f1 score
53 f1 = f1_score(y_test, model.predict(x_test))
54 # calculate precision
55 my_precision = precision_score(y_test, model.predict(x_test))
56 # calculate recall
57 my_recall = recall_score(y_test, model.predict(x_test))
58 # calculate ap
59 my_ap = average_precision_score(y_test, model.predict(x_test))
59

```

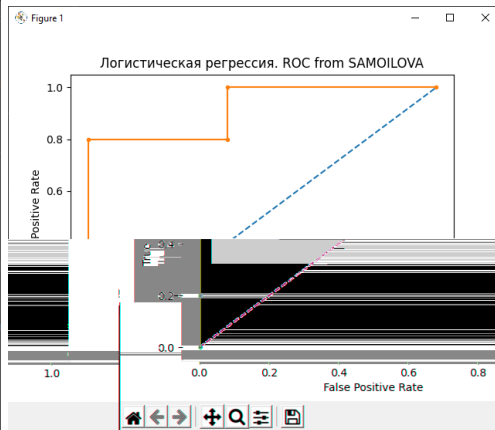
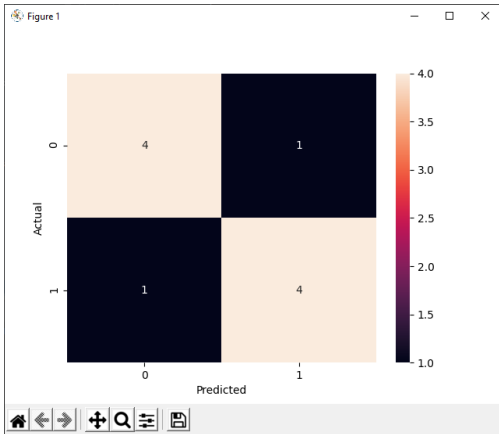
```

18 #names = f'age', 'sex'
19 dataframe = pd.read_csv('data.csv')
20
21 print('Введены параметры')
22 #обучаем модель
23 print('Формат обучения')
24 print(dataframe.head())
25 array = dataframe.values
26 X = array[:,0:3]
27 Y = array[:,3]
28 x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.3, random_state=0)
29 #создаем экземпляр
30 model = AdaBoostClassifier()
31 #обучаем модель
32 model.fit(x_train, y_train)
33 #предсказываем вероятности
34 probs = model.predict_proba(x_test)
35 print(model.predict(x_test))
36 print(probs)
37 # keep probabilities from the raw predicted results
38 probs = probs[:,1]
39 # calculate AUC
40 my_auc = roc_auc_score(y_test, probs)
41 print('AUC: %.3f' % my_auc)
42 # calculate roc curve
43 fpr, tpr, thresholds = roc_curve(y_test, probs)
44 # plot no skill
45 plt.plot([0, 1], [0, 1], 'r--')
46 # plot the roc curve for the model
47 plt.plot(fpr, tpr, 'b-')
48 plt.title('Receiver Operating Characteristic from SAMOILOVA')
49 plt.ylabel('True Positive Rate')
50 plt.xlabel('False Positive Rate')
51 plt.show()
52 # calculate f1 score
53 f1 = f1_score(y_test, model.predict(x_test))
54 # calculate precision
55 my_precision = precision_score(y_test, model.predict(x_test))
56 # calculate recall
57 my_recall = recall_score(y_test, model.predict(x_test))
58 # calculate ap
59 my_ap = average_precision_score(y_test, model.predict(x_test))
59

```

И 3 и вкклв л пфквбв й влблблй ил в пфкв з лж нв нв вб. и лнвпйш нв нв вб
й и вкклй л пфквбб .
Д б квв.

f- ROC AUC



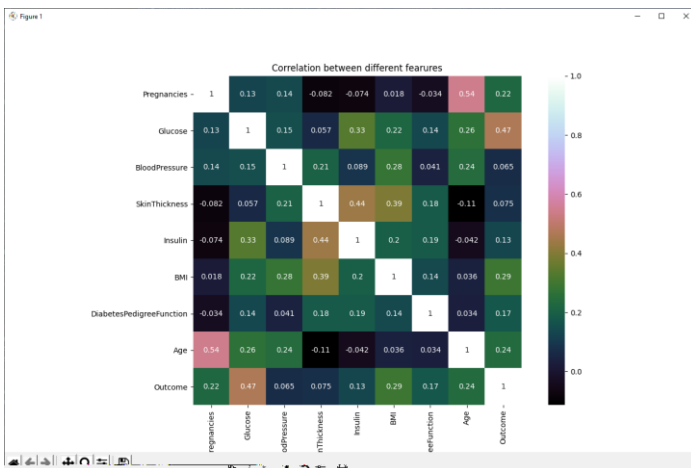
```
[[0.3018346 0.6981654 ]
 [0.69176147 0.30823853]
 [0.05322031 0.94677969]
 [0.20417163 0.79582837]
 [0.58169393 0.41830607]
 [0.88906961 0.11093039]
 [0.02920068 0.97079932]
 [0.11033348 0.88966652]
 [0.67029328 0.32970672]
 [0.12638422 0.87361578]]
```

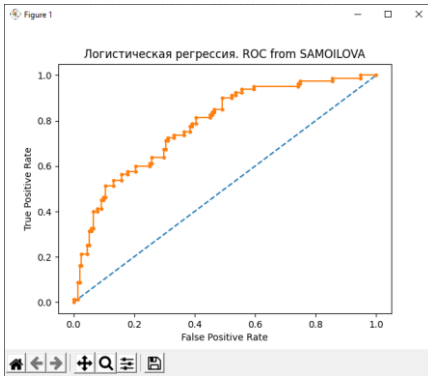
Accuracy: 0.8

	precision	recall	f1-score	support
0	0.80	0.80	0.80	5
1	0.80	0.80	0.80	5
accuracy			0.80	10
macro avg	0.80	0.80	0.80	10
weighted avg	0.80	0.80	0.80	10

AUC: 0.920

Д б квв 2





Name: Outcome, Length: 537, dtype: int64

	Pregnancies	Glucose	Insulin	BMI	Age
334	1	95	58	23.9	22
139	5	105	325	36.9	28
485	0	135	250	42.3	24
547	4	131	166	33.1	28
18	1	103	83	43.3	33
..
71	5	139	140	28.6	26
106	1	96	0	22.4	27
270	10	101	0	45.6	38
435	0	141	0	42.4	29
102	0	125	0	22.5	21

[537 rows x 5 columns]

	precision	recall	f1-score	support
0	0.76	0.87	0.81	151
1	0.66	0.49	0.56	80
accuracy			0.74	231
macro avg	0.71	0.68	0.69	231
weighted avg	0.73	0.74	0.72	231

Д б квв0+

Д б квв4.

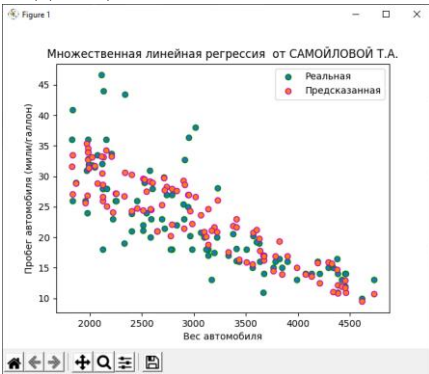
LinearRegression.

<https://archive.ics.uci.edu/ml/datasets/Auto+MPG>

('cylinders', 'displacement', 'horsepower', 'weight', 'acceleration', 'model_year', 'origin', 'car_name'):

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.
- 8.

ВДНЦ ПЦ7



G:/PyCharm_tatsamoilova/sklearn_simple/REGR/My_Auto_Lin_Regr.py
RMSE= 4.069215579662049

Process finished with exit code 0

Д б квв 5.

LinearRegression,

Ш ЛНЗ б ивбл квъ

DataFrame: wallsMaterial, floorNumber, floorsTotal, totalArea, kitchenArea, latitude, longitude, price

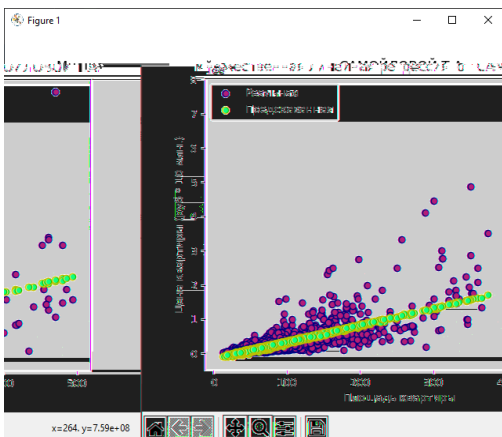
з плнш нв нв бб7

DataFrame: floorNumber, floorsTotal, totalArea, kitchenArea.

price

moscow dataset 2020.csv

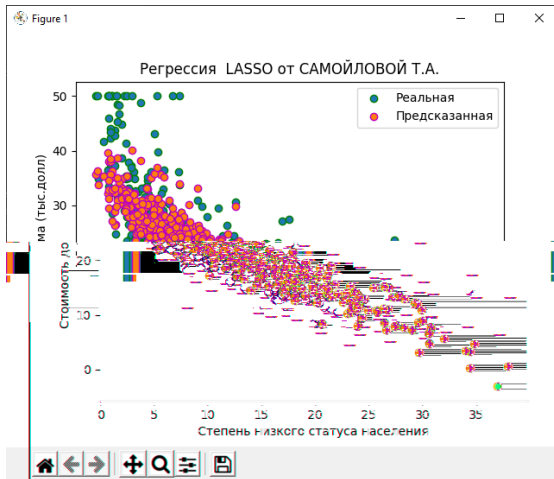
-2019.



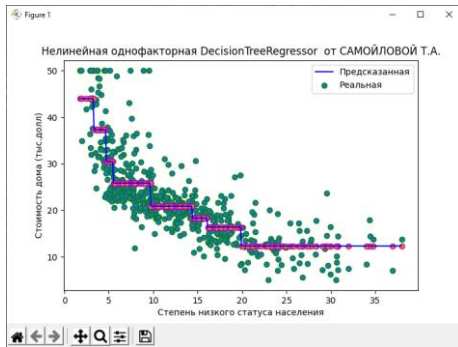
```
wallsMaterial floorNumber floorsTotal ... latitude longitude price
0 brick 1 5.0 ... 55.723379 37.628577 5600000
1 brick 1 5.0 ... 55.725980 37.671031 4650000
2 brick 1 5.0 ... 55.735976 37.657817 2990000
3 brick 1 7.0 ... 55.786698 37.595321 4390000
4 brick 2 5.0 ... 55.767894 37.665920 4890000
[5 rows x 8 columns]
RMSE= 16059592.911492676
```

Д б квв 6.

LassoCV

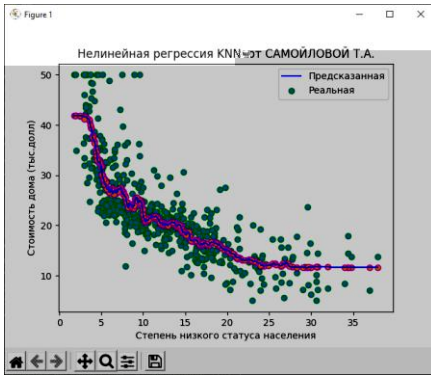


Д б квв 7



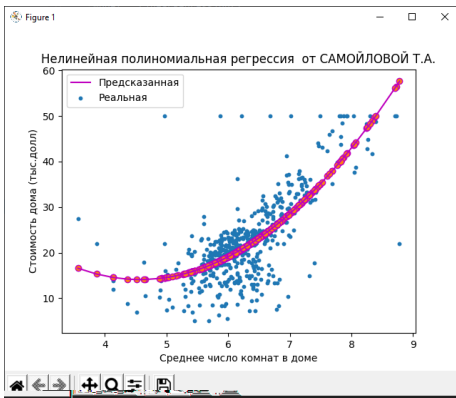
Д б квв 8

KNeighborsRegressor



rmse= 5.089057644349284
 - r2 0.6932172724449991

Д б квв 9
LinearRegression



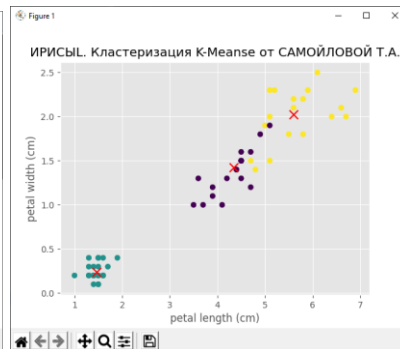
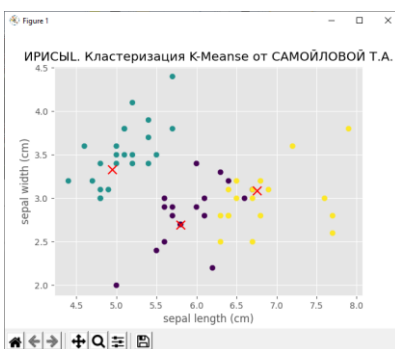
Д б квв 10.

Repository - <http://archive.ics.uci.edu/ml/>

r2_score

- mean_squared_error (MSE),

И 4 и лнвпйшзи пвнвд vbb й и вкклй л пфвквв .
Д б квв 1 K-Meanse



1 1 1 0 0 2 1 1 0 2 2 0 2 0 2 0 1 2 0 1 1 1 0]

1 2 0

1 1 1 2 0 0 1 1 0 2 2 0 2 0 2 0 1 2 0 1 1 1 0]

- accuracy_score = 0.9333333333333333

- F-

	precision	recall	f1-score	support
0	0.89	0.89	0.89	19
1	1.00	1.00	1.00	23
2	0.89	0.89	0.89	18
accuracy			0.93	60
macro avg	0.93	0.93	0.93	60
weighted avg	0.93	0.93	0.93	60

Д б квв .

Spending Score (1-

:

CustomerID,Genre,Age,Annual Income (k\$),Spending Score (1-100)

0001,Male,19,15,39

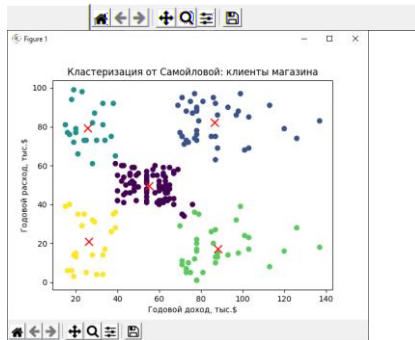
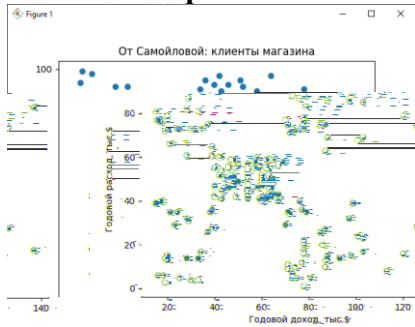
0002,Male,21,15,81

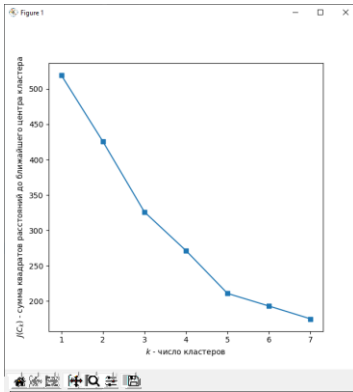
0003,Female,20,16,6

0004,Female,23,16,77

д б вйл пв

бквнубб лпфб и зи пвнл .





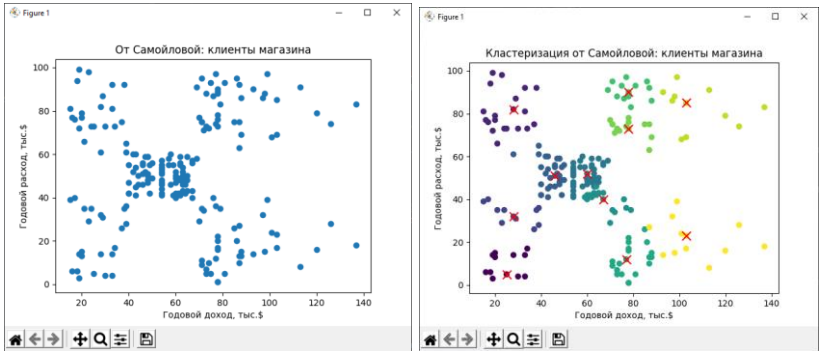
```
[4 2 4 2 4 2 4 2 4 2 4 2 4 2 4 2 4 2 4 2 4 2 4 2 4 2 4 2 4 2 4 2 4
2 4 2 4 2 4 0 4 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 3 1 0 1 3 1 3 1 0 1 3 1 3 1 3 1 3 1 0 1 3 1 3 1
3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3
1 3 1 3 1 3 1 3 1 3 1 3 1]
```

Д б квв3 AffinityPropagation (<https://habr.com/ru/post/321216/>)

Genre, Age, Annual Income (k\$), Spending Score (1-100)].

нвбкв лзл бvбвкп

вип .



```
[2 1 0 1 2 1 0 1 0 1 0 1 0 1 2 1 2 1 2 1 0 1
0 1 2 3 2 1 0 1 0 1 0 1 0 1 2 1 2 1 2 3 2 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 5 4 5 5 4
4 4 5 4 5 4 4 4 4 5 4 4 5 4 4 4 5 5 4 4 5 4 5 4
4 5 4 7 5 8 5 7 6 8 6 8 5 8 6 7 6 8 6 8 6 7 5 7
6 7 5 8 6 7 6 7 6 8 6 7 6 8 6 8 5 7 6 7 6 8 6 7
10 8 6 8 6 7 6 7 6 8 10 9 10 9 10 9 10 9 10 9 10 9
10 9 10 9 10 9 10 9]
```

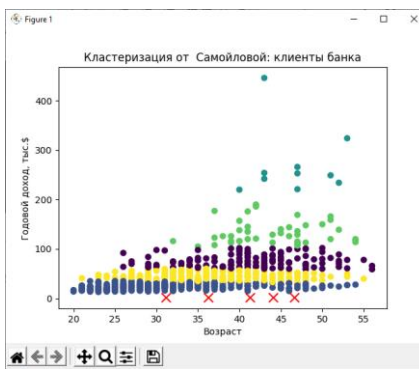
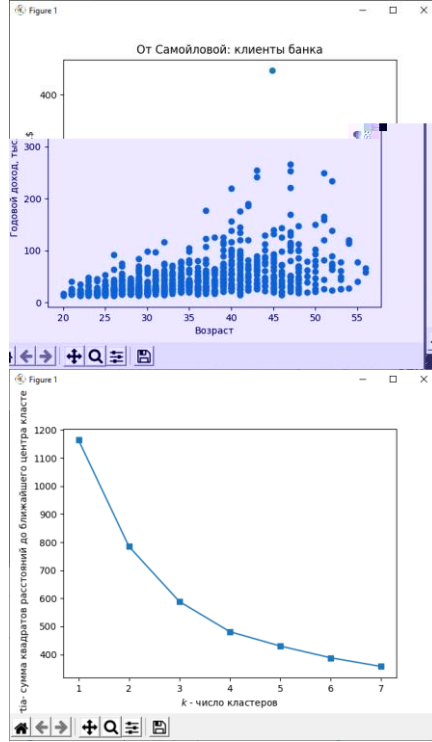
Д б квв4.

Num, Customer Id, Age, Edu, Years Employed, Income, Card Debt, Other Debt, Defaulted, DebtIncomeRatio
0,1,41,2,6,19,0.124,1.073,0.0,6.3

1,2,47,1,26,100,4.582,8.218,0.0,12.8
 2,3,33,2,10,57,6.1110000000000001,5.8020000000000005,1.0,20.9
 3,4,29,2,4,19,0.6809999999999999,0.516,0.0,6.3
 4,5,47,1,31,253,9.308,8.908,0.0,7.2

-
 -
 д в вйл пв

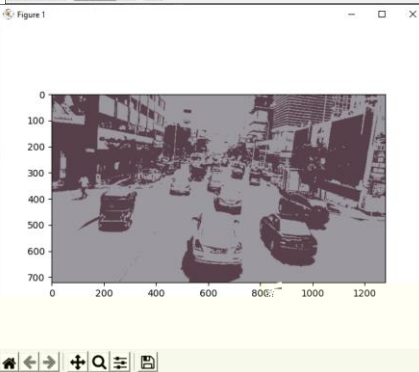
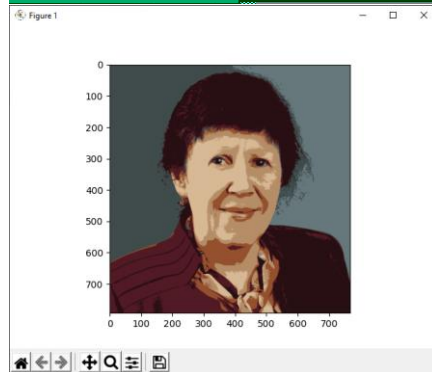
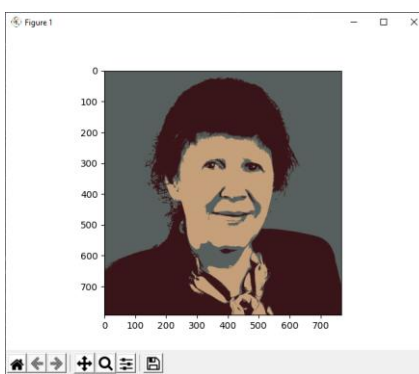
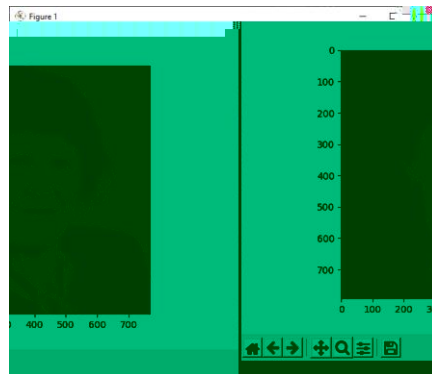
бквнвб лфв и зи пвнл .

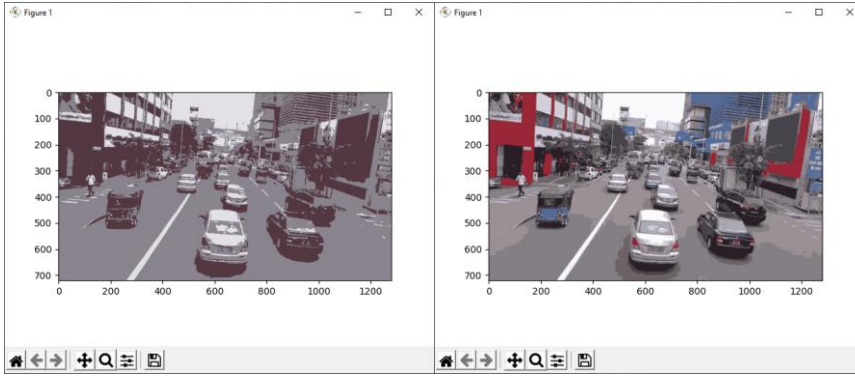


Д б квв 5

К-Meanse

CV

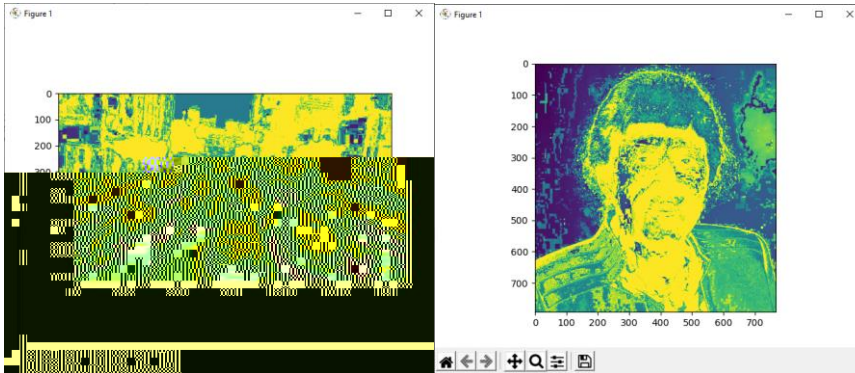




Д б квв 6

DBSCAN

sklearn.cluster



Д б квв 7.

-

бквнубб лшфб и зи пвнл .
Д б квв 8

д б вйл пв

И 5 Квжллкшв впр й и вкклй л прквбб. вквпрфв з вв и лнрпйш .
Д б квв.. python-keras TensorFlow.

<https://archive.ics.uci.edu/ml/datasets/diabetas>

5. 2-

)

Д б квв .

-

<https://archive.ics.uci.edu/ml/datasets/Auto+MPG>

('cylinders', 'displacement', 'horsepower', 'weight', 'acceleration', 'model_year', 'origin', 'car_name'):

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.
- 8.

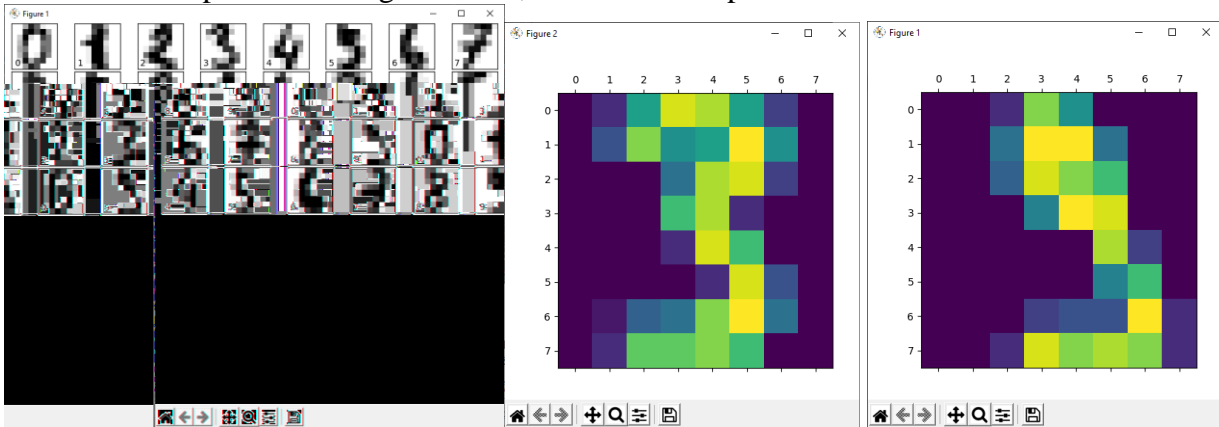
Д б кбв 0.

MLPClassifier

MNIST

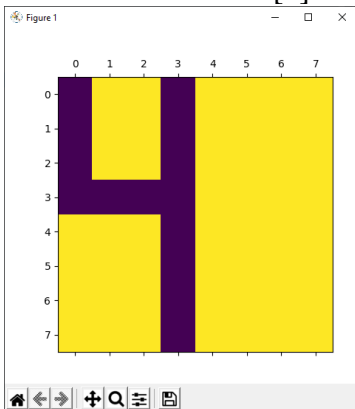
We have 1797 samples

Number of samples in training set: 1437, number of samples in test set: 360



- [3]

- [9]



- [4]

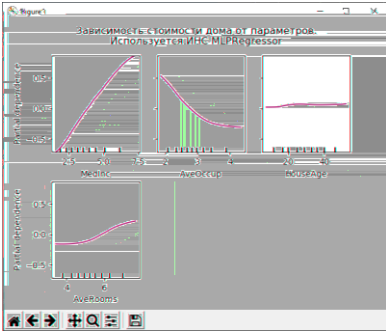
Д б кбв 4.

MLPClassifier

https://scikit-learn.org/stable/auto_examples/inspection/plot_partial_dependence.html#california-housing-data-preprocessing

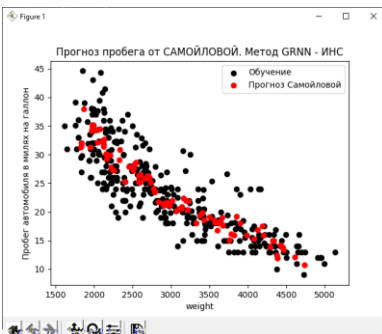
MedInc, HouseAge, AveRooms, AveBedrms, Population, AveOccup, Latitude Longitude.

Target -



к ивд н в3л +

Д б квв 2+



[[11.86674329]
[15.98118317]

[21.34850449]
[12.45185296]]
ОК!!

Д б квв 6

Д б квв 6.
WindowsForm -

AForge .NET

AForge

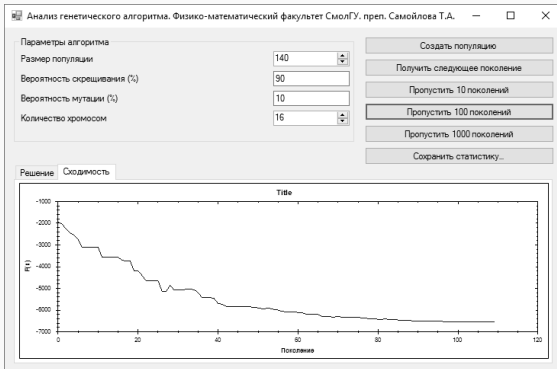
Д б квв 7.

Chart
Accord .NET

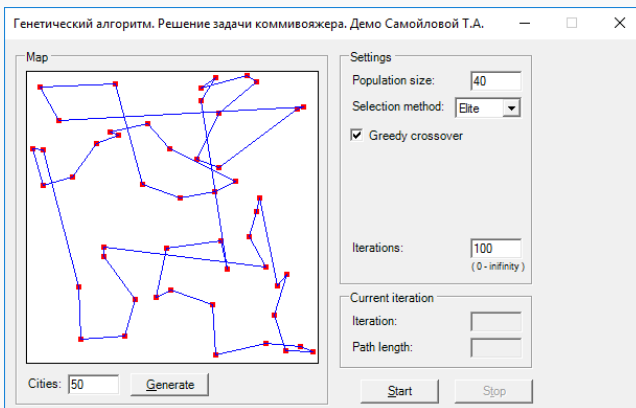
Accord.Genetic

Excel

Д б квв 8.



Д б квв 9.



И 3 Квфвиз вв й влбш й и вккл лл пфвквь .
Д б квв . . scikit-fuzzy

Fuzzy C-Means

Fuzzy partition coefficient (FPC).

• random - (x,y)

•

•

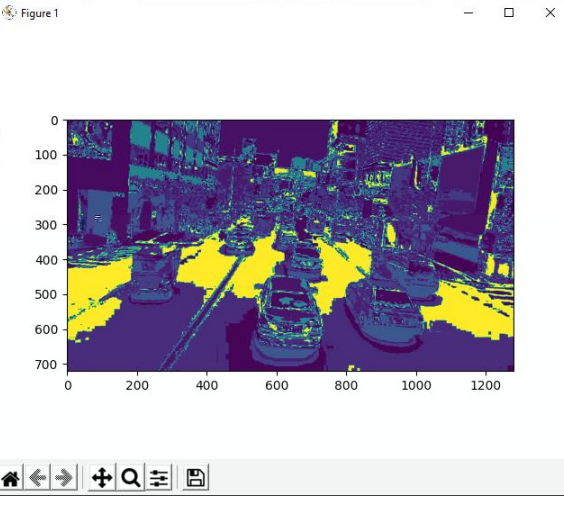
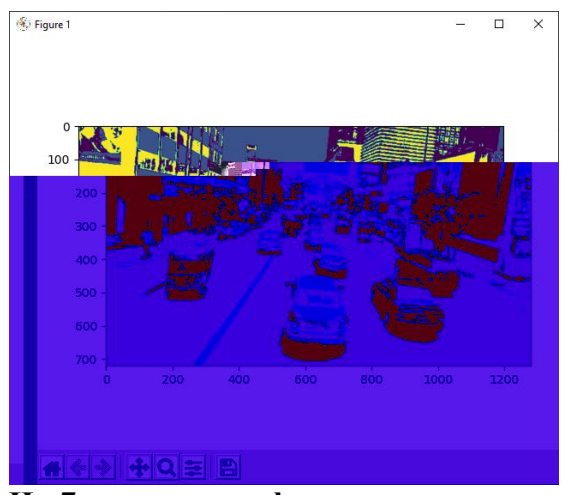
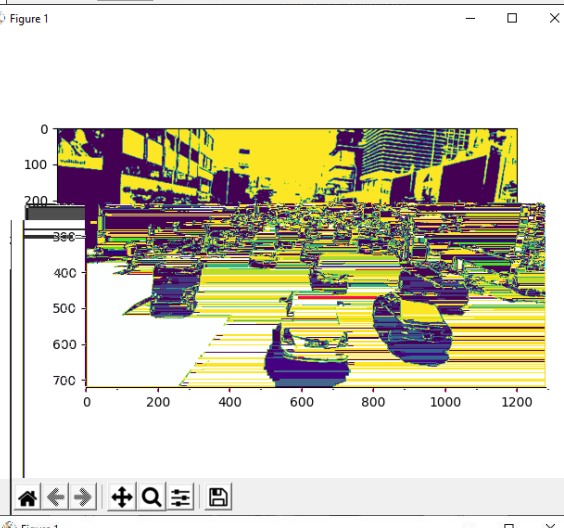
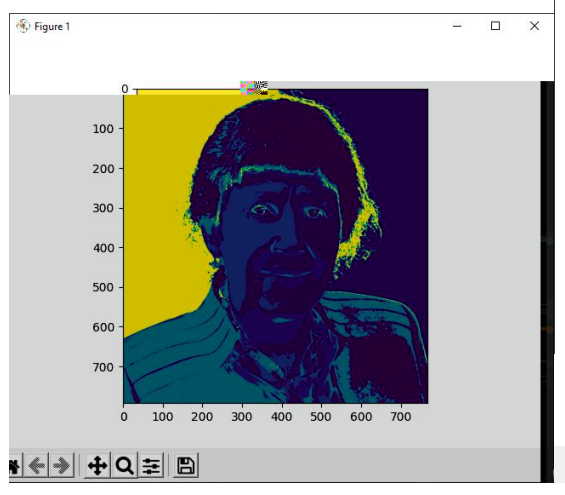
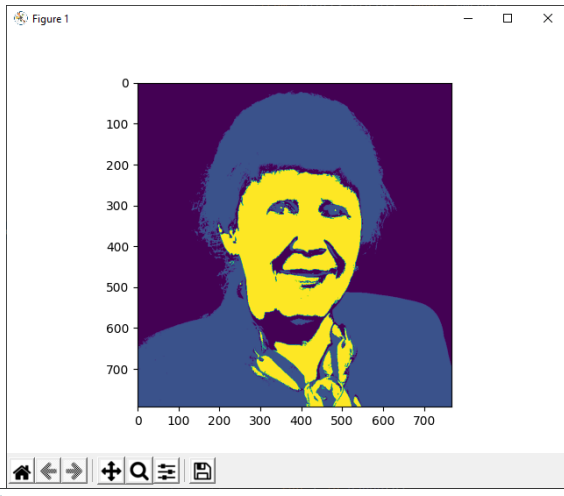
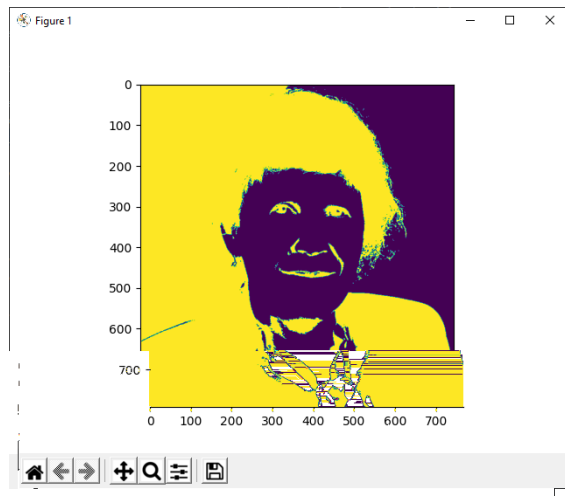
•

FPC)

Д 6 квв 2

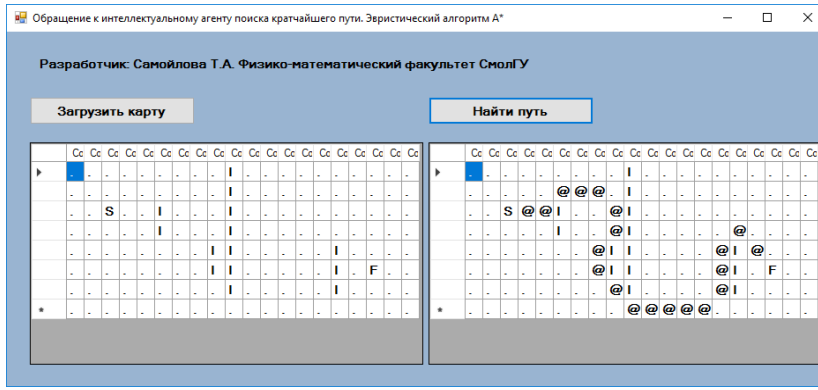
K-Means

scikit-fuzzy

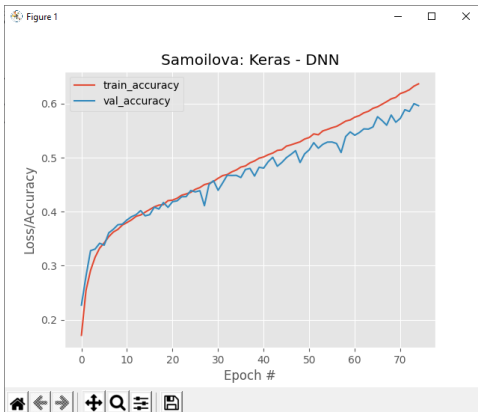
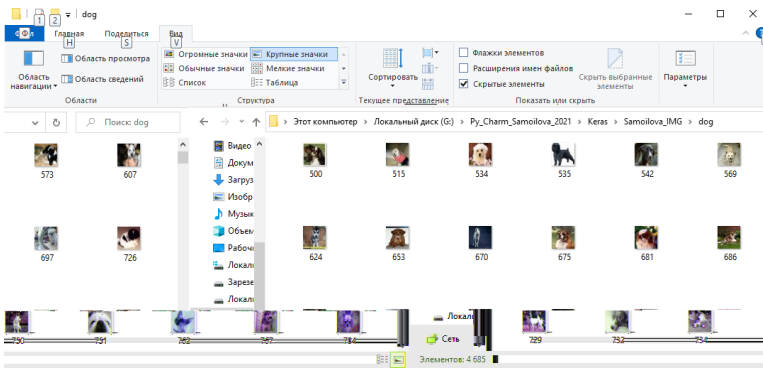


И 7 дн лпз вквивз пт ицк л вкп .
Д 6 квв+
WEBAP-

Windows Form -



И 8 З лй мщпвнквл днвкбв .
Д б квв 1.



```
[Loading images...
Загрузила рисунки
Epoch 1/75
1172/1172 [=====] - 17s 14ms/step - loss: 2.2387 - accuracy: 0.1713 -
val loss: 2.1468 - val accuracy: 0.2271
Epoch 2/75
1172/1172 [=====] - 18s 15ms/step - loss: 2.0790 - accuracy: 0.2540 -
val loss: 2.0104 - val accuracy: 0.2804
Epoch 3/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.9818 - accuracy: 0.2912 -
val loss: 1.9385 - val accuracy: 0.3279
Epoch 4/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.9275 - accuracy: 0.3157 -
val loss: 1.9023 - val accuracy: 0.3309
Epoch 5/75
```

```

Epoch 72/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.0821 - accuracy: 0.6217 -
val_loss: 1.1715 - val_accuracy: 0.5888
Epoch 73/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.0719 - accuracy: 0.6258 -
val_loss: 1.1702 - val_accuracy: 0.5856
Epoch 74/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.0566 - accuracy: 0.6325 -
val_loss: 1.1427 - val_accuracy: 0.6000
Epoch 75/75
1172/1172 [=====] - 17s 14ms/step - loss: 1.0427 - accuracy: 0.6367 -
val_loss: 1.1614 - val accuracy: 0.5962
[Evaluating network...
      precision    recall  f1-score   support

Samoilova_IMG/airplane      0.69      0.57      0.62      1250
Samoilova_IMG/bird          0.44      0.71      0.54      1317
Samoilova_IMG/car           0.83      0.57      0.68      1274
Samoilova_IMG/cat           0.56      0.39      0.46      1240
Samoilova_IMG/deer          0.51      0.52      0.51      1254
Samoilova_IMG/dog           0.59      0.45      0.51      1146
Samoilova_IMG/frog          0.61      0.65      0.63      1277
Samoilova_IMG/horse         0.79      0.57      0.66      1235
Samoilova_IMG/ship          0.56      0.82      0.66      1281
Samoilova_IMG/truck         0.65      0.69      0.67      1226

   accuracy
 macro avg      0.62      0.59      0.59      12500
 weighted avg   0.62      0.60      0.60      12500

Serializing network and label binarizer...
Process finished with exit code 0

```

Дб кбв .



```

[INFO] loading network and label binarizer...
Samoilova_IMG/dog

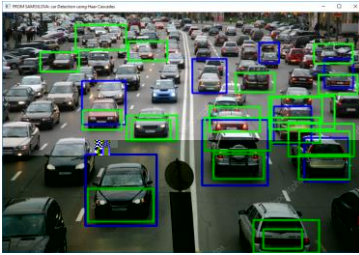
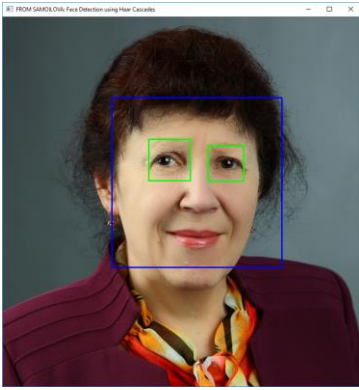
```

Дб кбв0.

Python-

OpenCV

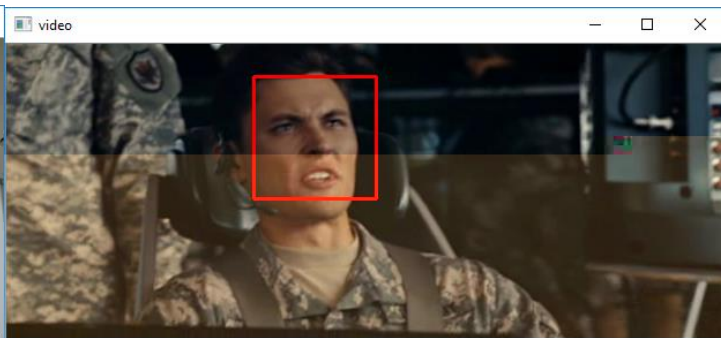
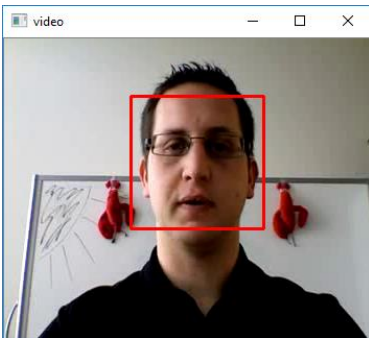
XML -



Д б квв 1.
ббвмппз

Python-

- XML -

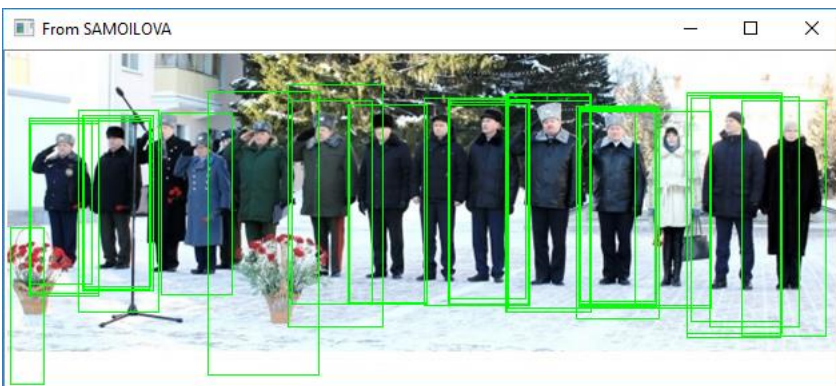


Д б квв 5.

python-

opencv

HOG,



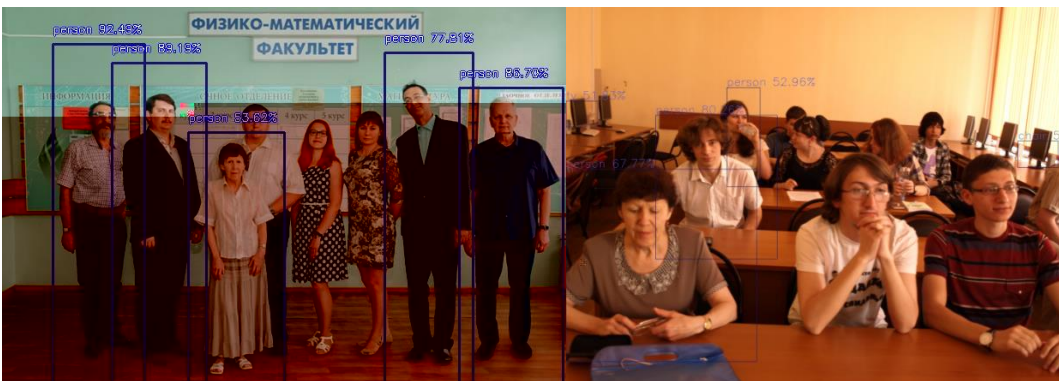
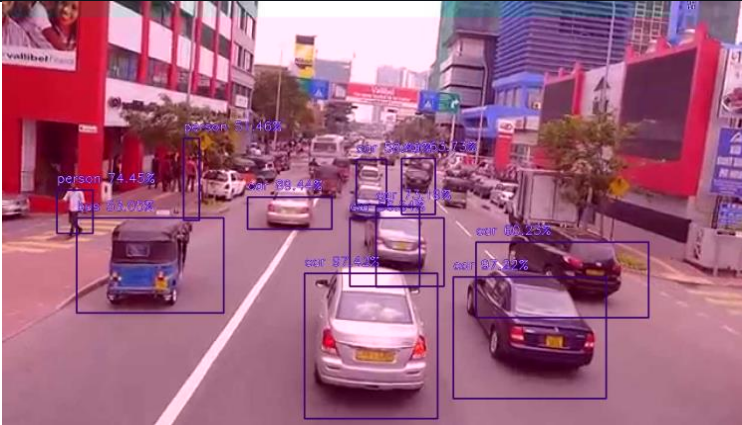
YOLO - yolo-tiny.h5.

1:

```

car : 56.54352307319641
bus : 53.0504584312439
car : 60.22621989250183
car : 97.22371101379395
car : 97.42370843887329
person : 51.46095156669617
car : 58.49268436431885
car : 63.7304425239563
person : 74.44759607315063
car : 89.44361805915833
car : 73.18670153617859

```



Л й л пльвищк ь н лп

-
-

Пвий ш биь й л пльвищк л л в д ф к в ь

- 1.
- 2.
- 3.
- 4.
- 5.

б+з нрпвнрр лувкр крв нвдпищ пл л лвкрв бр врмиркш йлбпир
 з+лвкклркшв нвр п в з нрпвнрр лувкр крв бир пвр пц вж ппв п вбр

Пвр пц ь з лкпнлицк ь н лп

2.

з нрпвнрр лувкр крв пвр пц вж з лкпнлицк лжн лпш

1.

		*)
1		
2		

(*)

2.

1		4,75-5
2		3,75-4,5
3		3-3,5
4		

Д б крв бир и лн пнкшт д крврж

з нрпвнрр лувкр крв шмликврв и лн пнкшт н лп

1.

		*)
1		
2		

2.

з+лвкклркшв нвр п в з нрпвнрр лувкр крв бир мнлйвг пплркжж ппв п вбр

лмнл ш з з д йвкт

1.

2.

3.

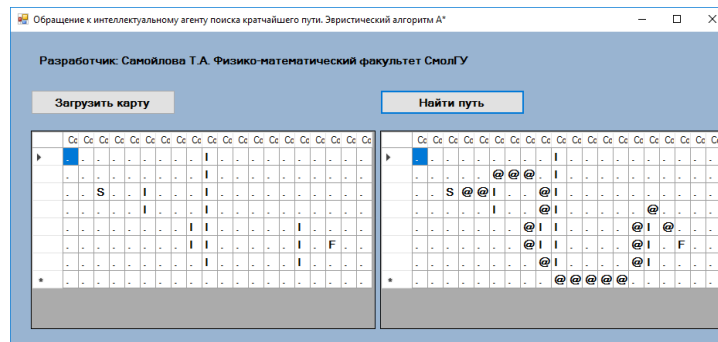
4.

5. ROC-
- 6.
- 7.
- 8.
- 9.
- 10.
- 11.
- 12.
- 13.
- 14.
- 15.
- 16.
- 17.
- 18.
- 19.

(k-

Л н дву з д й вк в блккл л бивп

- 1.
- 2.



З н п в н в б л в к в к в л ш в п к з д й в к в

1.

1		
2		
3		

(*)

2.

1		4,75-5
2		3,75-4,5
3		3-3,5
4		

**4+ вивіркі кл кж в блмлік в пвищк ж пфв кл жи в пвн пнш
4+ +Л кл к ь ир пвн пнн**

1. /
2022. 243 ISBN 978-5-534-01042-8. URL:
<https://urait.ru/bcode/490020>
2.
243
ISBN 978-5-534-07818-3. URL: <https://urait.ru/bcode/494506>
3. 2- :
157 ISBN 978-5-534-
11361-7. URL: <https://urait.ru/bcode/494434>
4. /
2- 157
ISBN 978-5-534-07467-3. URL : <https://urait.ru/bcode/490657>
5.
257 ISBN 978-5-9916-
8250-3. URL: <https://urait.ru/bcode/490259>
6.
250 ISBN 978-5-9916-
8251-0. URL: <https://urait.ru/bcode/471000>
7. /
256
ISBN 978-5-534-14916-6. URL: <https://urait.ru/bcode/485440>
8.
93 ISBN 978-
5-534-07819-0. URL : <https://urait.ru/bcode/494505>
9.
91
ISBN 978-5-534-00551-6. URL : <https://urait.ru/bcode/492094>
10.
278 ISBN 978-5-534-00734-3. URL:
<https://urait.ru/bcode/490386>
11. 85 ISBN 978-5-534-15561-7.
URL: <https://urait.ru/bcode/508804>
12.
332
ISBN 978-5-534-13619-7. URL: <https://urait.ru/bcode/497448>
13. 397 ISBN 978-5-534-11659-5. URL:
<https://urait.ru/bcode/495988>

4+ +Блмлік в пвищк ь ир пвн пнн

1. -Hill, 1997. 414 p.

2. URL: [http:// www. machinelearning.ru/ wiki/images/6/6d/Voron-ML-1.pdf](http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf)
- 3.
- 4.
5. 2014.
- 6.
- 7.
- 8.

7.3. Внефкщнв пн л бк лнй вблккл-пвнвлй йпкбз вблкклж впр кпвнквп

1. moodle.smolgu.ru).
2. intuit.ru).
3. opened.ru).
4.).
5. - (<http://window.edu.ru>).
6. ([http:// http://msdn.microsoft.com](http://msdn.microsoft.com)).

8+ пвнв ицкл-пвткбфв злвл в мвфкбв

224 12

9. нлн ййквл л в мвфкбв

- 1.
2. 9
3. Python 3.9
4. PyCharm Pro
- 5.

