Project - Classify Clothes from Fashion MNIST Dataset using Machine Learning Techniques

100% completed

12 Concepts | 7 Questions | 11 Assessments | 1,193 Learners

Python | scikit-learn | Pandas | Data Processing | Machine Learning | Dimensionality Reduction | Hyperparameter Tuning | Free | Guided Project

Welcome to this project on Classify Clothes from Fashion MNIST Dataset with a couple of Machine Learning algorithms like SGD Classifier, XGBClassifier, Softmax Regression (multi-class LogisticRegression), DecisionTreeClassifier, RandomForestClassifier, Ensemble (with soft voting) using scikit-learn. In this project, you will use Python and scikit-learn to build Machine Learning models, and apply them to predict the class of clothes from Fashion MNIST Dataset.

In this end-to-end Machine Learning project, you will get a hands-on overview of how to methodologically solve a machine learning classification problem. As a part of it, you will understand various methods of improvising the models using hyperparameter tuning, dimensionality reduction using the corresponding scikit-learn classes. You will also evaluate the performance of your final ensembling model using various performance metrics.

Skills you will develop:

1. scikit-learn
2. Machine Learning
3. Hyperparameter Tuning
4. Dimensionality Reduction
5. Python Programming
6. Ensemble modeling
7. Data Preprocessing
8. Pandas

Attempted by

and 1,189 more

Shivam Pankaj Singh Susanta Nath Kunal

This topic is part of below listed courses -

Machine Learning Specialization (Includes Deep Learning) | Machine Learning | Data Science Specialization | Data Science Specialization with TF2 by CloudxLab | Machine Learning Specialization with TF2 - CloudxLab | Data Science Specialization with TF2 by E&ICT, IITR

Instructor:

Sandeep Giri
Founder, CloudxLab.com

START NOW

End to End ML Project - Fashion MNIST - Description

Objective

Fashion-MNIST is a dataset of Zalando's article images —consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label.

The objective of the project is - to use Fashion-MNIST data set to identify (predict) different fashion products(articles) from the given images using Machine Learning.

We will be following the below steps to solve this problem:

1. Importing the libraries
2. Using some pre-defined utility functions
3. Loading the data
4. Cleaning the data
5. Dividing the dataset into training and test dataset using train_test_split in the ratio 85:15
6. Training several models and analyzing their performance to select a model
7. Use dimensionality reduction to improve the 'training', 'fine-tuning' and 'prediction' time.
8. Fine-tuning the model by finding the best hyper-parameters and features
9. Evaluating selected model using test dataset

Acknowledgements

Cloudxlab is using this "Fashion MNIST" problem for its machine learning learners for learning and practicing. Fashion-MNIST dataset is a collection of fashion article's images provided by Zalando . We thank Zalando Research for hosting the dataset.

Mark as Completed
Request Certificate
Index
Next

15 Comments

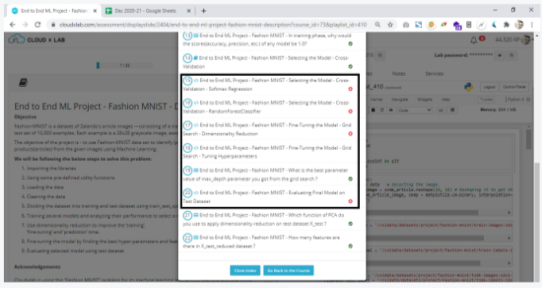
Unfollow conversation

Add comment

Sakti Prasad Pattanayak 3 days ago

Dear Sir,

After reviewing several times, couldn;t run the code.



Upvote Reply Share

Rajtilak Bhattacharjee 3 days ago

Hi,

Please schedule a meeting with me sometime tomorrow from the below link:

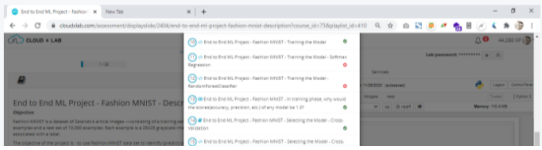
https://rajtilak.youcanbook.me/

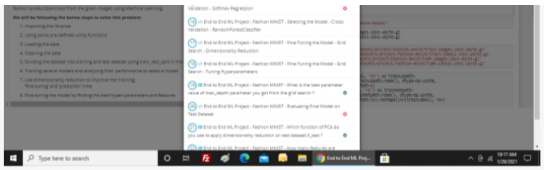
You will receive an invite with a Hangout link. Let's meet over Hangout and discuss this problem.

Thanks.

Upvote Reply Share

Sakti Prasad Pattanayak 3 days ago





Dear Sir,

There are few things couldn't fix kindly check the error sign.

Upvote Reply Share

Rajtilak Bhattacharjee 3 days ago

Hi,

Please start from step 1, check the answer, match it against your code, make any changes if necessary, and then move to the next step to repeat the same process.

Thanks.

Upvote Reply Share

Abhishek Gaurav 4 months ago

i have completed project - Fashion MNIST but when my page is not openning by default. It is showing another page "project_fashion_mnist_410". Pls suggest and send my file location.

Upvote Reply Share

Abhishek Gaurav 4 months ago

where is saved my project - Fashion MNIST?

Upvote Reply Share

Sandeep Akode 4 months ago

It should be in ~/cloudxlab_jupyter_notebooks/Python.ipynb

Upvote Reply Share

Neeta Sahay 5 months ago

Hi ,

my lab subscription has expired. How to proceed with the course ?

Upvote Reply Share

Himanshu Rathod 5 months ago

Hi,

You can buy Lab Subscription separately here - https://cloudxlab.com/lab/

Thanks,

Upvote Reply Share

Cyril George 6 months ago

i cant find the data. how to download the dataset from which git repository?

Upvote Reply Share

Rajtilak Bhattacharjee 6 months ago

Hi,

You do not need to worry, as you move ahead in this project, we will show you how to access the data and from where.

Thanks.

Upvote Reply Share

Vivek Bohra 8 months ago

Hi,
To do this project, I wanted to open a new notebook on right side window. But I am not getting any option. Whenever I open a file or create new one, Jupiter always open in new tab of browser.
Please guide how to open in right pan along with the actions defined so that I can easily submit the solution for verification and mark completed.
Thanks,
Vivek

Upvote Reply Share

CloudxLab 8 months ago

Hi,

The notebooks on the right side are default notebooks and cannot be replaced as they are used by the assessment engine. What you can do is to save the existing notebook by going to File -> Download as -> .ipynb format and save it with some other name, then delete the content of the default notebook and start afresh.

Thanks.

-- Rajtilak Bhattacharjee

Upvote Reply Share

Avishek Desarkar 10 months ago

i cant find the data set pls resolve ASAP, how to download the dataset from which git repository

Upvote Reply Share

CloudxLab 10 months ago

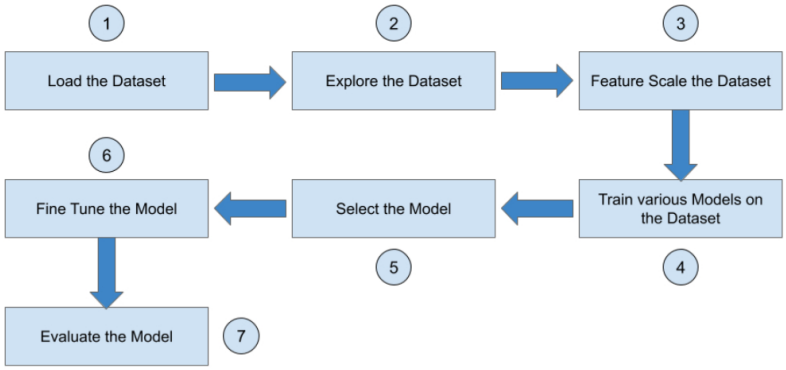
Hi @avishekdesarkar:disqus ,

Can you please follow the next steps?

Upvote Reply Share

End to End ML Project - Fashion MNIST - Step 1 - Load the Dataset

First, we will load the Fashion MNIST dataset.



Mark as Completed

Request Certificate

Previous Index Next

Be the first one to comment!

Add comment

End to End ML Project - Fashion MNIST - Importing Libraries

To start with the code, we would need some libraries. Please import the same into the environment, as per instructions mentioned below in Instructions box.

INSTRUCTIONS

Please import the required libraries as mentioned below:

```
numpy as np
pandas as pd
gzip
matplotlib
matplotlib's pyplot as plt
```

Please include the below line as it is.

```
%matplotlib inline
```

Please complete the code to import these libraries.

Submit Answer See Answer Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Previous Index Next

19 Comments

Unfollow conversation

Add comment

Sakti Prasad Pattanayak 3 days ago

Dear Sir,

I tried this way from one cell to another, still error presence while running the code, please check and revert, as time is running out to obtain the certificates, please help.

"2 4 9 10 11 15, 6 17 18 20"

Upvote Reply Share

Rajtilak Bhattacharjee 3 days ago

Hi,

Let me assure you that the answers are 100% correct and have been tested multiple times before we made them available to our users. Delete all the existing code from the notebook and copy paste only the code from the answer, this will address the issues you are facing.

Thanks.

Upvote Reply Share

Sandeep Sathyamurthy 3 months ago

Hi, i am getting an error for numpy import.But, the code syntax is correct. I have followed the steps of restarting the server through control panel and resubmit.still the same.Can you please help.

code:

import numpy as np
import pandas as pd
import gzip
%matplotlib inline
import matplotlib
import matplotlib.pyplot as plt

Upvote Reply Share

Rajtilak Bhattacharjee 3 months ago

Hi,

Please share a screenshot of your code and the error that you are getting.

Thanks.

Upvote Reply Share

Sandeep Sathyamurthy 3 months ago

Hi Rajtilak,

It is working fine now,thanks

Upvote Reply Share

RONIT ROY 5 months ago

What is the use of gzip library?

What is the need of the syntax %matplotlib inline

1 Upvote Reply Share

Rajtilak Bhattacharjee 5 months ago

Hi,

This module provides a simple interface to compress and decompress files just like the GNU programs **gzip** and **gunzip** would. The data compression is provided by the `zlib` module.

%**matplotlib inline** sets the backend of **matplotlib** to the '**inline**' backend: With this backend, the output of plotting commands is displayed **inline** within frontends like the Jupyter notebook, directly below the code cell that produced it. The resulting plots will then also be stored in the notebook document.

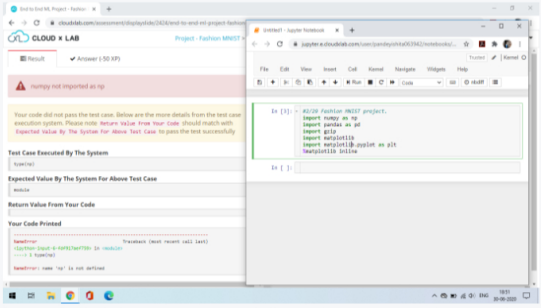
Thanks.

Upvote Reply Share

ishita pandey 7 months ago

Hi,

Will my code be accepted if I create a new notebook and run it? I did so and error showed as shown in screenshot.



Upvote Reply Share

Rajtilak Bhattacharjee 7 months ago

Hi,

You need to use the default notebook on the right side of the screen. Please try to restart your server by following all steps from the below link to resolve your issue:

https://discuss.cloudxlab.com/t/im-having-problem-with-assessment-engine-how-should-i-fix/3734

Thanks.

Upvote Reply Share

Archisman Chatterjee 8 months ago

It would have been good if we could do this project in a fresh Jupyter file.
Already the notepad is bloated with past assessments and projects and it becomes difficult to keep track.

Upvote Reply Share

CloudxLab 8 months ago

Hi,

You can download or use Save As to save your existing notebook, delete all the existing content from it, and then start afresh with this new project.
Hope that helps.

Thanks.

-- Rajtilak Bhattacharjee

Upvote Reply Share

Rahul Singh 9 months ago

@disqus_XTh3bUKOBh:disqus My lab subscription has expired but I have Jupyter notebook installed on my laptop. So, is it necessary to renew the lab subscription to complete this project??

Upvote Reply Share

CloudxLab 9 months ago

Hi Rahul,

This project is automatically evaluated by the playground Jupyter notebook. That functionality will not be available in your local Jupyter installation. So it is necessary to have a lab subscription to complete this project.

Thanks.

-- Rajtilak Bhattacharjee

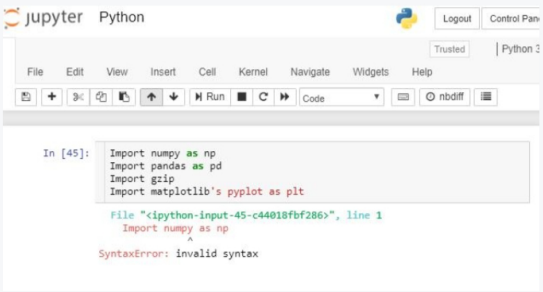
Upvote Reply Share

Mohit Agarwal 9 months ago

@cloudx please get the kernel and backend function properly. even the basic codes spell out the error?

Upvote Reply Share

Mohit Agarwal 9 months ago



@C@disqus_AHPvi4aTC8:disqus Cloudx please get the kernel and backend function properly. even the basic codes spell out the error?

Upvote Reply Share

CloudxLab 9 months ago

Hi Mohit,

Import is a keyword and almost all keywords starts lower case in Python. So please amend your code as follows:

import numpy as np

This should work.

Thanks

Thanks.

-- Rajtilak Bhattacharjee

Upvote Reply Share

Ashish Gupta a year ago

i have imported all the necessary libraries
import numpy as np

i am getting this error

numpy not imported as np

Upvote Reply Share

Mohit Agarwal 9 months ago

import numpy as np
import pandas as pd
import gzip
%matplotlib inline
import matplotlib
import matplotlib. pyplot as plt

Upvote Reply Share

CloudxLab 9 months ago

Hi Mohit,

Could you please share a screenshot of your updated code and the error that you are getting.

Thanks.

-- Rajtilak Bhattacharjee

Upvote Reply Share

End to End ML Project - Fashion MNIST - Defining Functions

Let us define some functions which will be used in this course.

The showImage() function given below displays the input dataset instance (image) on the screen.

INSTRUCTIONS

Copy paste the below function as is:

```
def showImage(data):
    some_article = data   # Selecting the image.
    some_article_image = some_article.reshape(28, 28) # Reshaping it to get the 28x28 pixels
    plt.imshow(some_article_image, cmap = matplotlib.cm.binary, interpolation="nearest")
    plt.axis("off")
    plt.show()
```

Mark as Completed Get Hint Request Certificate

Previous Index Next

Be the first one to comment!

End to End ML Project - Fashion MNIST - Loading the data

Let us load the Fashion MNIST dataset from Cloudxlab's below mentioned folder location (this dataset is copied from Zalando Research repository).

Location - '/cxldata/datasets/project/fashion-mnist/'

You need to load the below 4 dataset files:

1. train-images-idx3-ubyte.gz - this contains the Training dataset
2. train-labels-idx1-ubyte.gz - this contains the Training labels (target dataset)
3. t10k-images-idx3-ubyte.gz - this contains the Test dataset
4. t10k-labels-idx1-ubyte.gz - this contains the Test labels

The class labels for Fashion MNIST are:

```
Label   Description

  0       T-shirt/top
  1       Trouser
  2       Pullover
  3       Dress
  4       Coat
  5       Sandal
  6       Shirt
  7       Sneaker
  8       Bag
  9       Ankle boot
```

Out datasets consists of 60,000 images and each image has 784 features. An image consists of 28x28 pixels, and each pixel is a value from 0 to 255 describing the pixel intensity. 0 for white and 255 for black.

INSTRUCTIONS

Please define following string variables to store the location path of the dataset files. The dataset file location path should contain the file name also (appended in the end of the path).

The below variable contains location path for Training dataset

```
filePath_train_set = << your code comes here >>
```

The below variable contains location path for Training labels (target dataset)

```
filePath_train_label = << your code comes here >>
```

The below variable contains location path for Test dataset

```
filePath_test_set = << your code comes here >>
```

The below variable contains location path for Test labels

```
filePath_test_label = << your code comes here >>
```

Please create variables - (trainLabel, trainSet, testLabel, testSet) - using the below mentioned code. You can copy the below code as it is.

```
with gzip.open(filePath_train_label, 'rb') as trainLbpath:
    trainLabel = np.frombuffer(trainLbpath.read(), dtype=np.uint8,
                               offset=8)
with gzip.open(filePath_train_set, 'rb') as trainSetpath:
    trainSet = np.frombuffer(trainSetpath.read(), dtype=np.uint8,
                             offset=16).reshape(len(trainLabel), 784)

with gzip.open(filePath_test_label, 'rb') as testLbpath:
    testLabel = np.frombuffer(testLbpath.read(), dtype=np.uint8,
                              offset=8)

with gzip.open(filePath_test_set, 'rb') as testSetpath:
    testSet = np.frombuffer(testSetpath.read(), dtype=np.uint8,
                            offset=16).reshape(len(testLabel), 784)
```

trainLabel - contains Training label (target dataset)

trainSet - contains Training dataset

testLabel - contains Test label

testSet - contains Test dataset

Please copy the values of above created variables - trainSet, testSet, trainLabel and testLabel - in new variables - X_train, X_test, y_train and y_test respectively.

To get a feel of the data, you can view the article image at say index 0 of the Training dataset(X_train) and its corresponding label in the Target dataset (y_train). You can use showImage() function, that we defined earlier, for the same, e.g. showImage(X_train[0]).

Submit Answer See Answer Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Manmeet Kaur a month ago

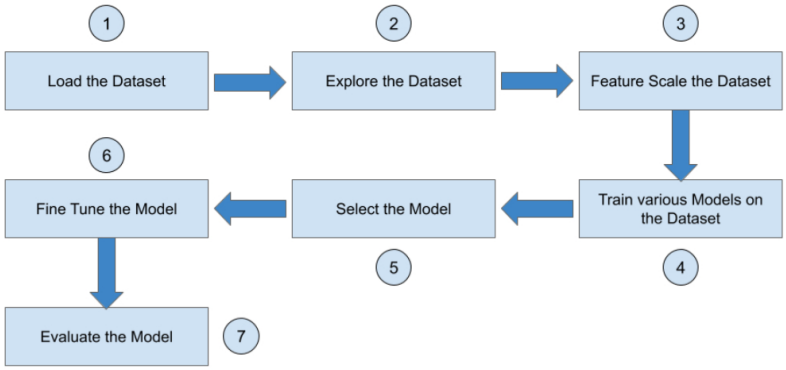
```
Getting following error on executing code:

FileNotFoundError: [Errno 2] No such file or directory: '/cxldata/datasets/project/fashion-mnist/train-images-idx3-ubyte.gz '
```

Though with following command it shows the files but when exceute rest of code,gets 'FileNotFoundError'

End to End ML Project - Fashion MNIST - Step 2 - Explore the Dataset

Next, we will explore the dataset we loaded.



Mark as Completed

Request Certificate

Previous Index Next

Be the first one to comment!

Add comment

Status: 7 / 30

How many instances are there in Fashion-MNIST Training dataset ?

- 70000
- 90000
- 60000

Submit Answer

Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Result

Correct answer!

Previous | Index | Next

Be the first one to comment!

Add comment

Status:
8 / 30

How many instances are there in Fashion-MNIST Test dataset ?

- 30000
- 10000
- 20000

Submit Answer

Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Result

Correct answer!

Previous Index Next

Be the first one to comment!

Add comment

Status:

9 / 30

How many features are there in Fashion-MNIST Training dataset ?

- 60000
- 70000
- 784
- 1

Submit Answer

Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Result

Correct answer!

Previous Index Next

Be the first one to comment!

R Add comment

End to End ML Project - Fashion MNIST - Understanding the data

Our training dataset consists of 60,000 images and each image has 784 features. An image consists of 28x28 pixels, and each pixel is a value from 0 to 255 describing the pixel intensity. 0 for white and 255 for black.

Let us have a look at one instance (an article image) of this training dataset X_train.

To view a single instance(an article image),all we need to do is grab an instance's feature vector, reshape it to a 28×28 array, and display it using Matplotlib's imshow() function.

Compare the digit in the image to its corresponding actual digit in the target dataset.

INSTRUCTIONS

Please use the showImage() created earlier to show image at X_train[0] as shown below

```
showImage(X_train[0])
```

Please check the corresponding article name (class) in target dataset (y_train[0]) if it matches the above image by using below code

```
y_train[0]
```

Mark as Completed | See Answer | Request Certificate

Previous | Index | Next

17 Comments

Unfollow conversation

Add comment

DILIP PRASAD 4 months ago

Hi team,

Why am I getting error like 'matplotlib' has no attribute 'imshow'?

My code is correct and started the assignments from the begining many times but getting stuck on this page.

I am using the default jupyter provided on the right side of the split screen. Sharing screen shot and the url as well for reference.

https://cloudxlab.com/assessment/displayslide/2446/end-to-end-ml-project-fashion-mnist-understanding-the-data?course_id=73&playlist_id=410

```
In [41]: showImage(X_train[0])
         y_train[0]

---------------------------------------------------------------------------
AttributeError                            Traceback (most recent c
all last)
<ipython-input-41-a4f9ca963ec6> in <module>
----> 1 showImage(X_train[0])
      2 y_train[0]

<ipython-input-21-ae4dfb063186> in showImage(data)
      2     some_article = data   # Selecting the image.
      3     some_article_image = some_article.reshape(28, 28) # Re
shaping it to get the 28x28 pixels
----> 4     plt.imshow(some_article_image, cmap = matplotlib.cm.bi
nary, interpolation="nearest")
      5     plt.axis("off")
      6     plt.show()

AttributeError: module 'matplotlib' has no attribute 'imshow'
```

Upvote Reply Share

Rajtilak Bhattacharjee 4 months ago

Hi,

imshow belongs to pyplot. Please use the below code:

```
import matplotlib.pyplot as plt
```

Thanks.

Upvote Reply Share

RONIT ROY 5 months ago

Why it is showing me error like name matplotlib is not defined though I have imported the required libraries?

Upvote Reply Share

Rajtilak Bhattacharjee 5 months ago

Hi,

Please share a screenshot of your code and the error that you are getting.

Thanks.

Upvote Reply Share

RONIT ROY 5 months ago

showImage(X_train[0])

Error:

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-9-78e84d826394> in <module>
----> 1 showImage(X_train[0])

<ipython-input-4-ae4dfb063186> in showImage(data)
      2     some_article = data   # Selecting the image.
      3     some_article_image = some_article.reshape(28, 28) # Reshaping it to get the 28x28 pixels
----> 4     plt.imshow(some_article_image, cmap = matplotlib.cm.binary, interpolation="nearest")
      5     plt.axis("off")
      6     plt.show()

NameError: name 'matplotlib' is not defined
```

Upvote Reply Share

Rajtilak Bhattacharjee 5 months ago

Hi,

As requested in my previous comment, please also share a screenshot of your code. Also, please let me know if you are writing this code on another tab or on the default Jupyter notebook provided on the right side of the split screen.

Thanks.

Upvote Reply Share

RONIT ROY 5 months ago

I have written the code in the right side of the split screen only,still it is not working.

Upvote Reply Share

Rajtilak Bhattacharjee 5 months ago

Hi,

As requested earlier, please share a screenshot of your entire screen with the code, and the error that you are getting.

Thanks.

Upvote Reply Share

Gandharv Bakshi 5 months ago

The Fashion MNIST that loads for me has apparel images - shoes, Tshirts etc. My code is below:

```
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import gzip
import os

%matplotlib inline
os.getcwd()
TRAIN_PATH = '/cxldata/datasets/project/fashion-mnist/'
filePath_train_set = os.path.join(TRAIN_PATH, "train-images-idx3-ubyte.gz")
filePath_train_label = os.path.join(TRAIN_PATH, "train-labels-idx1-ubyte.gz")
filePath_test_set = os.path.join(TRAIN_PATH, "t10k-images-idx3-ubyte.gz")
filePath_test_label = os.path.join(TRAIN_PATH, "t10k-labels-idx1-ubyte.gz")

with gzip.open(filePath_train_label, 'rb') as trainLbpath:
    trainLabel = np.frombuffer(trainLbpath.read(), dtype=np.uint8,
                               offset=8)
with gzip.open(filePath_train_set, 'rb') as trainSetpath:
    trainSet = np.frombuffer(trainSetpath.read(), dtype=np.uint8,
                             offset=16).reshape(len(trainLabel), 784)

with gzip.open(filePath_test_label, 'rb') as testLbpath:
    testLabel = np.frombuffer(testLbpath.read(), dtype=np.uint8,
                              offset=8)

with gzip.open(filePath_test_set, 'rb') as testSetpath:
    testSet = np.frombuffer(testSetpath.read(), dtype=np.uint8,
                            offset=16).reshape(len(testLabel), 784)

X_train = trainSet
X_test = testSet
y_train = trainLabel
y_test = testLabel

showImage(X_train[0])
```

Upvote Reply Share

Rajtilak Bhattacharjee 5 months ago

Hi,

Please submit your code through the assessment engine. Let me know if you face any challenges.

Thanks.

Upvote Reply Share

Gandharv Bakshi 5 months ago

Hi,

This isn't about the code. The challenge is that the text talks about the MNIST dataset being about handwritten digits. While the dataset is about T-shirts, Shoes and apparel.

Upvote Reply Share

Rajtilak Bhattacharjee 5 months ago

Hi,

Would suggest you to read the instructions carefully. This is not the MNIST dataset, this is the Fashion-MNIST dataset.

Thanks.

Upvote Reply Share

This comment has been removed.

Rajtilak Bhattacharjee 5 months ago

Hi,

The target variable is actually a digit representing the class of the clothes you are predicting. Also, I was not being sarcastic, if I came across like that please accept my sincere apologies.

Thanks.

Upvote Reply Share

Rajtilak Bhattacharjee 5 months ago

Hi,

Also, to continue my previous comment, I personally feel every comment has merit. There are no bad questions, or stupid questions. When I was learning Machine Learning, I too had my fair share of "stupid" questions, and today I feel that those questions were not stupid at all. Rather, they helped me clear some of the doubts that affect people for years to come.

Let me explain a bit on why I asked to observe the instructions more carefully. The first slide in this project explains the dataset, what is represents, what is consists of and other details. We do this with almost all of our guided projects because we feel without knowing the dataset, you will simply be performing the steps of the project but will not "learn" anything.

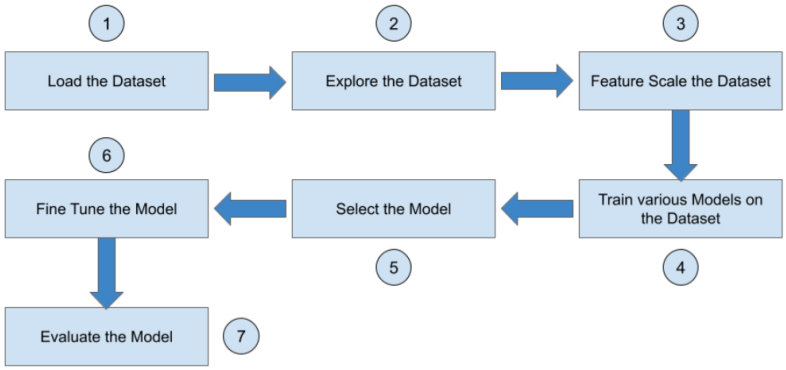
Once again, please accept my apologies if I sounded sarcastic. However, that was not my intention at all.

Thanks.

Upvote Reply Share

End to End ML Project - Fashion MNIST - Step 3 - Feature Scale the Dataset

Now we will use Feature Scaling on the dataset.



Mark as Completed
Request Certificate
Previous
Index
Next

End to End ML Project - Fashion MNIST - Data Preparation

Now, let us prepare data for training the model.

As part of data preparation, we need to perform following techniques on the data:

1. Shuffling
2. Feature Scaling

Shuffling the training dataset - to get uniform samples for cross validation

We need to shuffle our training data to ensure that we don't miss out any article (fashion product) in a cross validation fold.

Feature Scaling

Each image (instance) in the dataset has 784 pixels (features) and value of each feature(pixel) ranges from 0 to 255, and this range is too wide , hence we would need to use feature scaling here to apply standardization to this dataset X_train, so that all the values of each feature (pixel) is in a small range (based on the standard deviation value).

```
x_scaled = (x - x_mean) / standard deviation
```

Scaling is not needed for Decision Tree and Random Forest algorithms

INSTRUCTIONS

Please follow the below steps:

Create a random seed=42

```
np.random.seed(<<your code comes here>>)
```

Create shuffle indices of size 60000 (as we have 60000 images in the training dataset) and store it in a variable 'shuffle_index'

```
shuffle_index = np.random.permutation(<<your code comes here>>)
```

Shuffle the indices of X_train and y_train datasets by using 'shuffle_index' variable created above.

```
X_train, y_train = X_train[<<your code comes here>>], y_train[<<your code comes here>>]
```

Import StandardScaler from SKLearn's preprocessing

```
from <<your code comes here>> import StandardScaler
```

Create an instance of StandardScaler and store it in variable called 'scaler'

```
scaler = <<your code comes here>>
```

Apply standardization on training dataset X_train using the above created StandardScaler instance `scaler` using fit_transform method and store the scaled training dataset in X_train_scaled variable.

```
X_train_scaled = scaler.<<your code comes here>>(X_train.astype(np.float64))
```

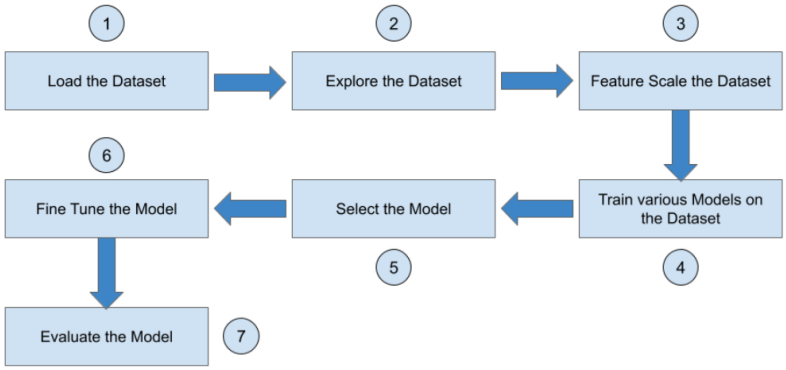
Submit Answer | Get Hint | See Answer | Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

34 Comments | Unfollow conversation

End to End ML Project - Fashion MNIST - Step 4 - Train various Models on the Dataset

Next, we will train various models on this dataset that we prepared.



End to End ML Project - Fashion MNIST - Training the Model

As the data preparation is over, let us now train a few Machine Learning(ML) models.

We will be training the following ML models:

Softmax Regression (multi-class LogisticRegression)

RandomForestClassifier

Ensemble (with soft voting)

INSTRUCTIONS

Please perform the below steps:

(1) Import accuracy_score from SKLearn

```
from <<your code comes here>> import accuracy_score
```

(2) Import precision_score and recall_score from SKLearn

```
from <<your code comes here>> import precision_score, recall_score
```

(3) Import f1_score from SKLearn

```
from <<your code comes here>> import f1_score
```

Submit Answer See Answer Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

7 Comments Unfollow conversation

End to End ML Project - Fashion MNIST - Training the Model - Softmax Regression

Let us now train the Softmax Regression (Logistic Regression - multi_class-multinomial). We will be doing the following as part of this exercise:

1. We will be first training the Softmax Regression (Logistic Regression - multi_class-multinomial) on the training dataset
2. Using the trained model, make the prediction on a sample instance and compare the prediction with the actual value.
3. Using the trained model, make the prediction on the whole training dataset
4. Calculate - accuracy, precision, recall and F1 Score for Softmax Regression (Logistic Regression - multi_class-multinomial).

INSTRUCTIONS

Please follow the below steps:

Import LogisticRegression from SKLearn

```
from <<your code comes here>> import  LogisticRegression
```

Create an instance of LogisticRegression by passing parameters - multi_class="multinomial", solver="lbfgs", C=10 and random_state=42 to the constructor and store this created instance in a variable called 'log_clf'.

```
# using Softmax Regression (multi-class classification problem)

log_clf = LogisticRegression(<<your code comes here>>)

# 'C' is hyprparameter for regularizing L2
# 'lbfgs' is Byoden-Fletcher-Goldfarb-Shanno(BFGS) algorithm
```

Now, train the model on 'scaled' training dataset

```
log_clf.<<your code comes here>>(X_train_scaled, <<your code comes here>>)
```

Make prediction on an instance from the training dataset (say instance at index '0' i.e. X_train[0]) using the above trained model 'log_clf', and store the predicted value in a variable called y_train_predict

```
y_train_predict = log_clf.<<your code comes here>>(X_train[0].reshape(1, -1))
```

Let us compare the actual value to the predicted value of the label. You can use showImage() function to see the image.

```
y_train[0]

y_train_predict[0]

showImage(X_train[0])
```

Make the predictions on the complete training dataset X_train_scaled using the above trained model 'log_clf' and save the result in variable 'y_train_predict'

```
y_train_predict = log_clf.<<your code comes here>>(X_train_scaled)
```

Calculate the various metrics scores like - accuracy, precision, recall, F1 score - using the actual and the predicted values and relevant functions, - and store them in respective variables - log_accuracy, log_precision, log_recall and log_f1_score.

```
log_accuracy = <<your code comes here>>(y_train, <<your code comes here>>)

log_precision = <<your code comes here>>(y_train, <<your code comes here>>, average='weighted')

log_recall = <<your code comes here>>(y_train, <<your code comes here>>, average='weighted')

log_f1_score = <<your code comes here>>(y_train, <<your code comes here>>, average='weighted')
```

You can print the above metrics values (accuracy, etc.) using the print() function

Submit Answer See Answer Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Previous Index Next

62 Comments

Unfollow conversation

Add comment

Sakti Prasad Pattanayak 4 days ago

Variable 'y_train_predict' not defined

Still confusion while entering the code given.

Upvote Reply Share

Rajtilak Bhattacharjee 3 days ago

Hi,

Either the variable y_train_predict was not defined in your code, or you need to run the previous cells from the beggining for them to take effect.

End to End ML Project - Fashion MNIST - Training the Model - RandomForestClassifier

Let us now train the RandomForestClassifier. We will be doing the following as part of this exercise:

1. We will be first training the RandomForestClassifier on the training dataset
2. Using the trained model, make the prediction on a sample instance and compare the prediction with the actual value.
3. Using the trained model, make the prediction on the whole training dataset
4. Calculate - accuracy, precision, recall and F1 Score for RandomForestClassifier.

INSTRUCTIONS

Please follow the below steps:

Import RandomForestClassifier from SKLearn

```
from <<your code comes here>> import RandomForestClassifier
```

Create an instance of RandomForestClassifier by passing parameters - n_estimators=20, max_depth=10, random_state=42, and store this created instance in a variable called 'rnd_clf'.

```
rnd_clf = RandomForestClassifier(<<your code comes here>>)

# Scaling is not needed for Decision Tree based algorithms like Random Forest and XGBoost
```

Now, train the model on training dataset

```
rnd_clf.<<your code comes here>>(X_train, <<your code comes here>>)
```

> **Note:** Please note that the training might take upto 2-3 minutes.

Make prediction on an instance from the training dataset (say instance at index '0' i.e. X_train[0]) using the above trained model 'rnd_clf', and store the predicted value in a variable called y_train_predict

```
y_train_predict = rnd_clf.<<your code comes here>>(X_train[0].reshape(1, -1))
```

Let us compare the actual value (digit) to the predicted value (digit). You can use showImage() function to see the image.

```
y_train[0]

y_train_predict[0]

showImage(X_train[0])
```

Make the predictions on the complete training dataset X_train using the above trained model 'rnd_clf' and save the result in variable 'y_train_predict'

```
y_train_predict = rnd_clf.<<your code comes here>>(X_train)
```

Calculate the various metrics scores like - accuracy, precision, recall, F1 score - using the actual and the predicted values and relevant functions, - and store them in respective variables - rnd_accuracy, rnd_precision, rnd_recall and rnd_f1_score.

```
rnd_accuracy = <<your code comes here>>(y_train, <<your code comes here>>)

rnd_precision = <<your code comes here>>(y_train, <<your code comes here>>, average='weighted')

rnd_recall = <<your code comes here>>(y_train, <<your code comes here>>, average='weighted')

rnd_f1_score = <<your code comes here>>(y_train, <<your code comes here>>, average='weighted')
```

You can print the above metrics values (accuracy, etc.) using the print() function

Submit Answer See Answer Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

23 Comments Unfollow conversation

Status:

17 / 30

In training phase, why would the scores(accuracy, precision, etc.) of any model be 1.0?

- Those are worst models for fashion-MNIST, they don't give good results(scores)
- Those are best models for fashion-MNIST, they give good results(scores)
- Those models have tendency to 'overfit'
- Those models have tendency to 'underfit'

Submit Answer

Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Result

✔ Correct answer!

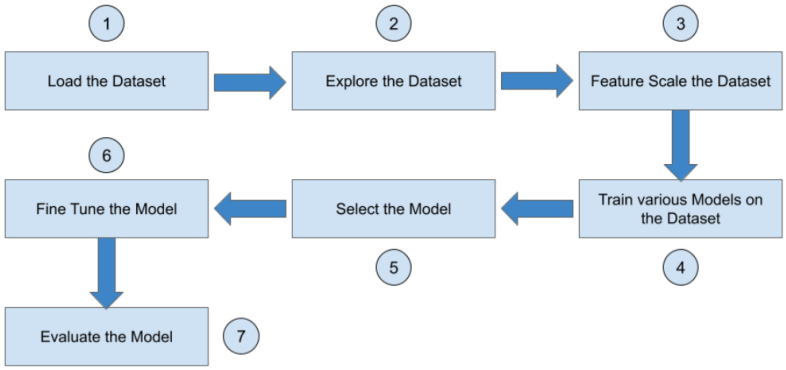
Previous | Index | Next

Be the first one to comment!

Add comment

End to End ML Project - Fashion MNIST - Step 5 - Select the Model

Now we will select the model.



Mark as Completed

Request Certificate

Previous Index Next

Be the first one to comment!

Add comment

End to End ML Project - Fashion MNIST - Selecting the Model - Cross-Validation

Now, let us use cross validation to find the proper score of each model, also to ensure that the model is not overfitting or underfitting. Based on this cross-validation, we will select the model for fine-tuning its hyperparameters.

NOTE:

- If the cross validation score values for a performance measure (say accuracy) are not varying significantly for various folds (k-folds), then we can say that the model is not overfitting.
- If the cross validation score values for a performance measure (say accuracy) are not very low for various folds (k-folds), then we can say that the model is not underfitting.

We will perform **k-fold cross-validation.** We will randomly split the training set into 3 distinct subsets called folds (**cv=3**). Since cross validation is a computing intensive and time consuming process, we are limiting 'cv' (no. of folds) to 3 instead of normally 10 folds. Then will train and evaluate each model 3 times by picking a different fold for evaluation every time and training on the other 2 folds The result will be an array containing the 3 evaluation scores for each of the measures - **accuracy, precision, recall, F1 score.** We will use **cross_val_score()** function to calculate **accuracy**

But accuracy is generally not the preferred performance measure for classifiers, especially when you are dealing with skewed datasets. (A dataset is said to be skewed when some classes are much more frequent than others.)

Even if the current training dataset may not be skewed, the future test dataset (live) on which the model runs can be skewed, hence, considering we may get skewed dataset in future, let us calculate Precision, Recall and F1 score also for the models. And will use **cross_val_predict()** function to create confusion matrix to calculate **Precision, Recall and F1 score.**

Mark as Completed
Request Certificate
Previous Index Next

End to End ML Project - Fashion MNIST - Selecting the Model - Cross-Validation - Softmax Regression

Let us import some libraries and define a function (display_scores()) which we will be using for cross-validation.

We will be performing k-fold cross-validation with 3 folds (cv=3) on the Softmax Regression model, and calculating the mean accuracy, precision, recall and F1 score values for the same.

INSTRUCTIONS

Please follow the below steps:

Import the module `cross_val_score` and `cross_val_predict` from `sklearn.model_selection`

```
from sklearn.model_selection import << your code comes here >>
```

Import the module `confusion_matrix` from `sklearn.metrics`.

```
from sklearn.metrics import << your code comes here >>
```

Define a function called display_scores() which should print the score value which is passed to it as argument, and also calculate and print the 'mean' and 'standard deviation' of this score.

```
def display_scores(scores):
     <<your code comes here>>
```

Please create an instance of LogisticRegression called log_clf by passing to it the parameters - multi_class="multinomial", solver="lbfgs", C=10 and random_state=42

```
log_clf = LogisticRegression(<<your code comes here>>)
```

Please call cross_val_score() function by passing following parameters to it - the model (log_clf), the scaled training dataset (X_train_scaled), y_train, cv=3 and scoring="accuracy" - and save the returned value in a variable called log_cv_scores.

Call display_scores() function, by passing to it the log_cv_scores variable, to calculate and display(print) the 'accuracy' score, the mean of the 'accuracy' score and the 'standard deviation' of the 'accuracy' score.

```
log_cv_scores = cross_val_score(<<your code comes here>>)
display_scores(log_cv_scores)
```

Call mean() method on log_cv_scores object to get the mean accuracy score and store this mean accuracy score in a variable log_cv_accuracy.

```
log_cv_accuracy = log_cv_scores.<<your code comes here>>
```

Please call cross_val_predict() function by passing following parameters to it - the model (log_clf), the scaled training dataset (X_train_scaled), y_train, cv=3 - and save the returned value in a variable called y_train_pred.

```
y_train_pred = cross_val_predict(<<your code comes here>>)
```

Compute the confusion matrix by using confusion_matrix() function

```
confusion_matrix(y_train, <<your code comes here>>)
```

Calculate the precision score by the using the precision_score() function

```
log_cv_precision = precision_score(y_train, <<your code comes here>>, average='weighted')
```

Calculate the recall score by the using the recall_score() function

```
log_cv_recall = recall_score(y_train, <<your code comes here>>, average='weighted')
```

Calculate the F1 score by the using the f1_score() function

```
log_cv_f1_score = f1_score(y_train, <<your code comes here>>, average='weighted')
```

Print the above calculated values of log_cv_accuracy, log_cv_precision, log_cv_recall , log_cv_f1_score

Submit Answer Get Hint Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

50 Comments Unfollow conversation

End to End ML Project - Fashion MNIST - Selecting the Model - Cross-Validation - RandomForestClassifier

We will be performing k-fold cross-validation with 3 folds (cv=3) on the RandomForestClassifier model, and calculating the mean accuracy, precision, recall and F1 score values for the same.

INSTRUCTIONS

Please follow the below steps:

Please create an instance of RandomForestClassifier called rnd_clf by passing to it the parameters - n_estimators=20, max_depth=10 and random_state=42

```
rnd_clf = RandomForestClassifier(<<your code comes here>>)
```

Please call cross_val_score() function by passing following parameters to it - the model (rnd_clf), the training dataset (X_train), y_train, cv=3 and scoring="accuracy" - and save the returned value in a variable called rnd_cv_scores.

Call display_scores() function, by passing to it the rnd_cv_scores variable, to calculate and display(print) the 'accuracy' score, the mean of the 'accuracy' score and the 'standard deviation' of the 'accuracy' score.

```
rnd_cv_scores = cross_val_score(<<your code comes here>>)
display_scores(rnd_cv_scores)
```

Call mean() method on rnd_cv_scores object to get the mean accuracy score and store this mean accuracy score in a variable rnd_cv_accuracy.

```
rnd_cv_accuracy = rnd_cv_scores.<<your code comes here>>
```

Please call cross_val_predict() function by passing following parameters to it - the model (rnd_clf), the training dataset (X_train), y_train, cv=3 - and save the returned value in a variable called y_train_pred.

```
y_train_pred = cross_val_predict(<<your code comes here>>)
```

Compute the confusion matrix by using confusion_matrix() function

```
confusion_matrix(y_train, <<your code comes here>>)
```

Calculate the precision score by the using the precision_score() function

```
rnd_cv_precision = precision_score(y_train, <<your code comes here>>, average='weighted')
```

Calculate the recall score by the using the recall_score() function

```
rnd_cv_recall = recall_score(y_train, <<your code comes here>>, average='weighted')
```

Calculate the F1 score by the using the f1_score() function

```
rnd_cv_f1_score = f1_score(y_train, <<your code comes here>>, average='weighted')
```

Print the above calculated values of rnd_cv_accuracy, rnd_cv_precision, rnd_cv_recall , rnd_cv_f1_score

Submit Answer Get Hint Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Previous Index Next

37 Comments

Unfollow conversation

Add comment

Jishnu Bhatt a month ago

Dear Sir/Madam

Please check my issue as described below

```
from sklearn.model_selection import cross_val_score, cross_val_predict
from sklearn.metrics import confusion_matrix

def display_scores(scores):
    print("Scores:", scores)
    print("Mean:", scores.mean())
    print("Standard deviation:", scores.std())
```

End to End ML Project - Fashion MNIST - Selecting the Model - Cross-Validation - The Conclusion

You can print the various metrics of each model the following way:

```
print("=== Softmax === ")
display_scores(log_cv_scores)
print("log_cv_accuracy:", log_cv_accuracy)
print("log_cv_precision:", log_cv_precision)
print("log_cv_recall:", log_cv_recall)
print("log_cv_f1_score:", log_cv_f1_score)

print("=== Random Forest === ")
display_scores(rnd_cv_scores)
print("rnd_cv_accuracy:", rnd_cv_accuracy)
print("rnd_cv_precision:", rnd_cv_precision)
print("rnd_cv_recall :", rnd_cv_recall )
print("rnd_cv_f1_score:", rnd_cv_f1_score)
```

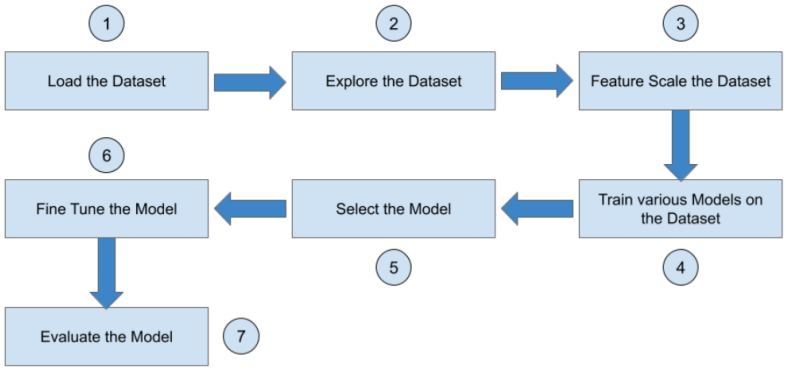
From the results of the cross-validation process, we see that both the logistic regression and random forest have given the best results (nearly accuracy - 85%, standard deviation for accuracy - 0.002, Precision, Recall, F1 score nearly 0.85).

Let us use Voting Classifier and proceed with the fine-tuning of the model (hyperparameters tuning).

Mark as Completed
Request Certificate
Previous Index Next

End to End ML Project - Fashion MNIST - Step 6 - Fine Tune the Model

Now we will fine tune our model.



Mark as Completed

Request Certificate

Previous Index Next

Be the first one to comment!

Add comment

End to End ML Project - Fashion MNIST - Fine-Tuning the Model - Grid Search - Dimensionality Reduction

We will perform using 'Grid Search' technique.

Grid search takes a lot of time on large datasets. Hence, let us apply 'Dimensionality Reduction' to the training dataset to reduce the number of features in the dataset, so that the time taken for grid search and prediction is reduced. Also, we will calculate the scores based on the reduced features.

We will also check, if dimensionality reduction leads to any significant loss of information from the images in our training dataset. If we get a significant loss of information with dimensionality reduction, we will not use dimensionality reduction for our training dataset (and hence the problem).

Our dataset is not like a Swiss-roll, therefore, we don't need to convert a 3-dimensional dataset to 2-dimensional plane, etc. Hence, we won't be using Manifold technique for dimensionality reduction here.

We will be using Projection technique (PCA) for dimensionality reduction for our problem.

We will use Scikit Learn's PCA class which uses SVD (Singular Value Decomposition) internally and also the projection.

You can experiment with various values of n_components (variance ratio).

For the current problem, with n_components=0.95, in the reduced dataset (X_train_reduced) we got only 187 features (out of original 784), and there was significant loss of information (quality) in the 'recovered' (decompressed) images. Hence, we have selected n_components=0.99, which gives 459 features (out of original 784) and there is no significant loss of information (quality) in the 'recovered' images.

The comparison of the 'original' dataset images and the 'compressed' dataset images (got after decompression) shows that there is not much information loss due to dimensionality reduction by using 0.99 variance ratio. Hence, we will go ahead with performing the Grid Search using this 'reduced' training dataset (X_train_reduced).

INSTRUCTIONS

For dimensionality reduction, please follow the below steps:

Import PCA from SKLearn

```
from <<your code comes here>> import PCA
```

Create an instance of PCA called 'pca', by passing to it the parameter n_components=0.99 (i.e. variance ratio of 0.99)

```
pca = PCA(<<your code comes here>>)
```

Apply PCA on the training dataset X_train dataset and save the result in a variable called X_train_reduced

```
X_train_reduced = pca.<<your code comes here>>(X_train)
```

Please check the number of components (features) present in the X_train_reduced dataset

```
pca.<<your code comes here>>
```

Please check if you have hit a total of 99% explained variance ratio with the select number of components:

```
np.sum(pca.<<your code comes here>>)
```

Please check if there is any loss of information due to dimensionality reduction. You can do this by recovering (decompressing) some of the images (instances) of X_train_reduced dataset.

Let us recover (decompress) some of the images (instances) of X_train_reduced dataset and check.

Please use inverse_transform function to decompress the compressed dataset (X_train_reduced) back to 784 dimensions , and save the resulting dataset in X_train_recovered variable.

```
X_train_recovered = pca.<<your code comes here>>(<<your code comes here>>)
```

Please use the below code and function as it is. It will display the original image and the compressed image (that was recovered after decompression).

```
import matplotlib
import matplotlib.pyplot as plt

def plot_digits(instances, images_per_row=5, **options):
    size = 28
    images_per_row = min(len(instances), images_per_row)
    images = [instance.reshape(size,size) for instance in instances]
    n_rows = (len(instances) - 1) // images_per_row + 1
    row_images = []
    n_empty = n_rows * images_per_row - len(instances)
    images.append(np.zeros((size, size * n_empty)))
    for row in range(n_rows):
        rimages = images[row * images_per_row : (row + 1) * images_per_row]
        row_images.append(np.concatenate(rimages, axis=1))
    image = np.concatenate(row_images, axis=0)
    plt.imshow(image, cmap = matplotlib.cm.binary, **options)
    plt.axis("off")

plt.figure(figsize=(7, 4))
plt.subplot(121)
# Plotting 'original' image
plot_digits(X_train[::2100])
plt.title("Original", fontsize=16)
plt.subplot(122)
# Plotting the corresponding 'recovered' image
plot_digits(X_train_recovered[::2100])
plt.title("Compressed", fontsize=16)
plt.show()
```

The comparison of the 'original' dataset images and the 'compressed' dataset images (got after decompression) shows that there is not much information loss due to dimensionality reduction by using 0.99 variance ratio.

Submit Answer | Get Hint | Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

18 Comments | Unfollow conversation

Add comment

Uday Narasimhamurthy Kandarpa 2 months ago

Also, your score engine is deducting -10XP each time i click for the same question (hint). Request you to validate and fix. For One question once we take the hint, next time we click hint to revalidate the code, it should not deduct 10XP.

Upvote Reply Share

Vagdevi K 2 months ago

Hi,

Understand your concern, but it is expected that the user understands and completes the code with minimal help from the Hint, which is why it deducts for each click. All the best!

Thanks.

Upvote Reply Share

Uday Narasimhamurthy Kandarpa 2 months ago

X_train_Recover is not getting defined due to unsupported operand type(s) for 'PCA' and 'float' . Based on previous comments, I checked my previous values, all of them were approved by the assessment engine, not sure why this error is coming. I took HINT as well, and compared / exected the code, still the same issue.

Upvote Reply Share

Vagdevi K 2 months ago

Hi,

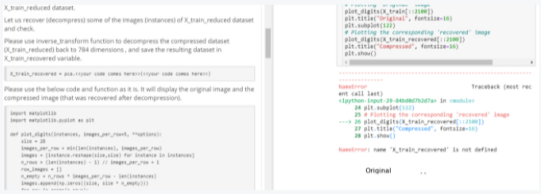
Would request you to share a screenshot of your code that you have written for X_train-recovered.

Thanks.

Upvote Reply Share

Khushboo Vyas 8 months ago

Help me with the code to define "X_train_recovered"



Upvote Reply Share

CloudxLab 8 months ago

Hi,

Would request you to share a screenshot of your code that you have written for X_train-recovered. Also, if you are stuck somewhere, you can take a hint or look at the answer.

Thanks.

-- Rajtilak Bhattacharjee

Upvote Reply Share

Piyush Jagtap 7 months ago

For me,using fit_transform method on pca worked instead of fit.

Upvote Reply Share

Rajtilak Bhattacharjee 7 months ago

Hi,

This discussion will help you understand the difference between fit and fit_transform:

https://stackoverflow.com/questions/23838056/what-is-the-difference-between-transform-and-fit-transform-in-sklearn

Thanks.

Upvote Reply Share

Rahul Singh 9 months ago

Hi!
In the previous learning item, we concluded that voting classifier is giving better results as compared to others and also decided to fine tune voting classifier model.
But in this, we are talking about XGBoost!! Please explain.

Upvote Reply Share

CloudxLab 9 months ago

Hi,

XGBoost with GridSearch will give much better result than fine tuning the voting classifier model. That is being depicted here.

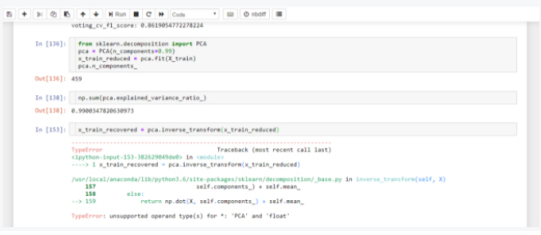
Thanks.

-- Rajtilak Bhattacharjee

Upvote Reply Share

Divya Kant Kumar 9 months ago

what type of this error



Upvote Reply Share

CloudxLab 9 months ago

Hi,

Would suggest you to go back to your previous codes and review them to check if they need any amendment. If you want, you can take a hint or look at the answer to match with your code and check if they need any changes.
Thanks.

-- Rajtilak Bhattacharjee

Upvote Reply Share

Dr. Manpreet Singh Sehgal a month ago

try

pca.fit_transform

in place of

pca.fit

Upvote Reply Share

Gopendra Mohan 10 months ago

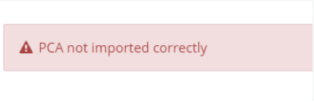
mean_squared_error(X_train,X_train_recovered)

Here , i am getting output as 56.381809623685456.
Please let me know the significance of this value . And is it good value or bad?

Upvote Reply Share

Ankur Sinha a year ago

code is running fine but as i submit, I get this annoying error!



End to End ML Project - Fashion MNIST - Fine-Tuning the Model - Grid Search - Tuning Hyperparameters

Let us now perform the Grid Search using the dimensionally reduced training dataset X_train_reduced.

Since our best model is Voting Classifier which is made up of two models Logistic Regression and Random Forrest. To do the grid search, we will have to supply the various values of parameters for both of the underlying models.

Since the grid search is a very intensive process, we are going to only try a handful of permutations because it would take a huge time otherwise.

NOTE:

In real-time scenarios, you might also like to train XGBoost and most likely, XGBoost would be the winning model. So, in those cases, you will have to do the hyperparameter training of XGBoost.

INSTRUCTIONS

Please follow the below steps:

Please import GridSearchCV from SKLearn

```
from <<your code comes here>> import GridSearchCV
```

For logistic regression, we are going to try the following parameters:

```
multi_class:["multinomial"], solver:["lbfgs"], C:[5],
```

Please note that there is one combination.

For Random Forrest, we are going to try the following parameters:

```
n_estimators:[20],
max_depth:[10, 15],
```

Please note that there are basically two combinations one with max_depth 10 and other with max_depth 15.

In the parameter grid, we need to prefix the name of the parameter with the name of model followed by double underscores.

Please fill the right values from the various parameters mentioned above:

```
param_grid = [
    {
        "lr__multi_class":["multinomial"],
        "lr__solver":["lbfgs"],
        "lr__C":[<< YOUR CODE GOES HERE>>],
        "rf__n_estimators":[20],
        "rf__max_depth":[<< YOUR CODE GOES HERE>>],
    }]
```

Please create an instance of LogisticRegression with the paramters: multi_class="multinomial", solver="lbfgs", C=10 , random_state=42 and assign it to log_clf_ens .

```
log_clf_ens = << YOUR CODE GOES HERE>>(<< YOUR CODE GOES HERE>>)
```

Please create an instance of RandomForestClassifier with the paramters: n_estimators=20, max_depth=10 , random_state=42 and assign it to rnd_clf_ens .

```
rnd_clf_ens = <<YOUR CODE GOES HERE>>(<<YOUR CODE GOES HERE>>)
```

Please create an instance of VotingClassifier as done earlier and assign it to voting_clf_grid_search:

```
voting_clf_grid_search = <<YOUR CODE GOES HERE>>(
    estimators=[('lr', log_clf_ens), ('rf', rnd_clf_ens)],
    voting='soft')
```

NOTE: Please note the name lr given to the logistic regression model and rf given to Random Forrest model.

We will perform the Grid Search with 3 folds i.e. cv=3. Please create an instance of GridSearchCV called 'grid search' by passing following parameter values - voting_clf_grid_search, param_grid, cv=3 and scoring='neg_mean_squared_error'

```
grid_search = GridSearchCV(<<your code comes here>>)
```

Run the grid search on the 'reduced' training dataset X_train_reduced

```
grid_search.<<your code comes here>>(X_train_reduced, y_train)
```

Get the best hyperparameter values

```
grid_search.<<your code comes here>>
```

Get the best estimator

```
grid_search.<<your code comes here>>
```

Let's look at the score of each hyperparameter combination used during the grid search

```
cvres = grid_search.cv_results_
for mean_score, params in zip(cvres["mean_test_score"], cvres["params"]):
    print(np.sqrt(-mean_score), params)
```

Submit Answer | Get Hint | Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Status:

26 / 30

What is the best parameter value of max_depth parameter you got from the grid search ?

- 100
- 120
- 10 to 15
- 0

Submit Answer

Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Result

Correct answer!

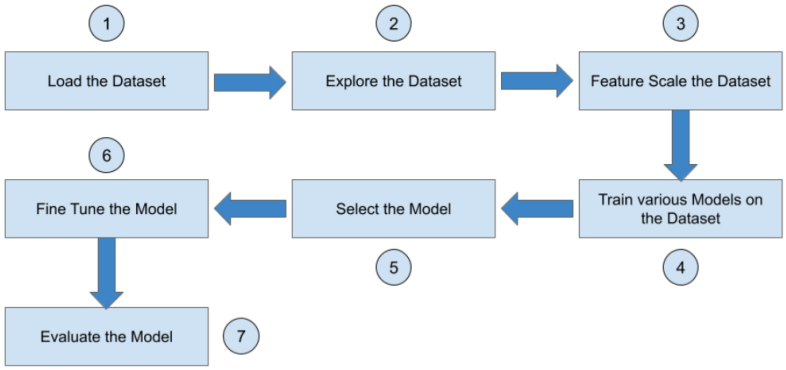
Previous Index Next

Be the first one to comment!

Add comment

End to End ML Project - Fashion MNIST - Step 7 - Evaluate the Model

Finally, we will evaluate our model.



End to End ML Project - Fashion MNIST - Evaluating Final Model on Test Dataset

Since, we already got our 'final' model from grid search (best_estimator_), let us evaluate the same on the test dataset.

Since, we performed grid search on the dimensionally reduced training dataset X_train_reduced, we need to apply dimensionality reduction the the test dataset also before we could use it for prediction on the test dataset.

INSTRUCTIONS

Please follow the below steps:

Store the best_estimator_ model, that we got from grid search, in a variable called final_model

```
final_model = grid_search.<<your code comes here>>
```

Import various score from sklearn's metrics package:

```
from <<your code comes here>> import accuracy_score
from <<your code comes here>> import confusion_matrix
from <<your code comes here>> import precision_score, recall_score
from <<your code comes here>> import f1_score
```

Remember, you have to use pca object of training dataset (you got on training dataset during dimensionality reduction (please don't create new instance of PCA) and only apply transform() on test dataset (not fit_transform).

Please apply transform() on X_test (using pca object) and store the resulting dataset in X_test_reduced variable

```
X_test_reduced = pca.<<your code comes here>>(X_test)
```

Perform the predictions on the X_test_reduced dataset using final model, and store the result in y_test_predict variable.

```
y_test_predict = final_model.<<your code comes here>>(X_test_reduced)
```

Create the confusion matrix

```
confusion_matrix(y_test, <<your code comes here>>)
```

Calculate various metrics scores like - accuracy, precision, recall, F1 score - using the actual and the predicted values and relevant functions, and store them in respective variables - final_accuracy, final_precision, final_recall and final_f1_score.

```
final_accuracy = <<your code comes here>>(y_test, <<your code comes here>>)
final_precision = <<your code comes here>>(y_test, <<your code comes here>>, average='weighted')
final_recall = <<your code comes here>>(y_test, <<your code comes here>>, average='weighted')
final_f1_score = <<your code comes here>>(y_test, <<your code comes here>>, average='weighted')
```

Print the values of final_accuracy, final_precision, final_recall and final_f1_score

Just check with a sample value, if the predictions were correct

```
y_test[0]

y_test_predict[0]

showImage(X_test[0])
```

Submit Answer Get Hint Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

38 Comments Unfollow conversation

Status:

29 / 30

Which function of PCA do you use to apply dimensionality reduction on test dataset X_test ?

- dim_reduce()
- dim_reduct()
- fit_transform()
- transform()

Submit Answer

Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Result

✔ Correct answer!

Previous | Index | Next

Be the first one to comment!

Add comment

Status:

30 / 30

How many features are there in X_test_reduced dataset ?

- 10000
- 784
- 459
- 187

Submit Answer

Request Certificate

Note - Having trouble with the assessment engine? Follow the steps listed here

Result

Correct answer!

Previous Index

Be the first one to comment!

Add comment