

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curriculum (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

085-assignment.ipynb x 081-working-with-apis.ipynb x 083-garch.ipynb x 082-test-driven.ipynb x data.py x model.py Python 3 (ipykernel) ○

Filter files by name

Name Last Modified

- ipymb_chec... 20 minutes ago
- images 2 months ago
- models 2 months ago
- .env 2 months ago
- 081-workin... 8 hours ago
- 082-test-dr... an hour ago
- 083-garch.ip... 2 months ago
- 084-model... 2 months ago
- 085-assign...** in a few seconds
- config.py 2 months ago
- data.py 9 hours ago
- main.py 2 months ago
- model.py 15 minutes ago
- stocks.sqlite seconds ago

8.5 Volatility Forecasting in South Africa ZA

In this assignment you'll build a model to predict stock volatility for the telecommunications company MTN Group.

Tip: There are some tasks in this assignment that you can complete by importing functions and classes you created for your app. Give it a try!

Warning: There are some tasks in this assignment where there is an extra code block that will transform your work into a submission that's compatible with the grader. Be sure to run those cells and inspect the submission before you submit to the grader.

```
[1]: %load_ext autoreload
%autoreload 2

import wqet_grader
from arch.univariate.base import ARCHModelResult

wqet_grader.init("Project 8 Assessment")
```

```
[2]: # Import your libraries here

import os
import sqlite3
from glob import glob

import joblib
import pandas as pd
import requests
import wqet_grader
```

Simple 0 2 3 5 6 Python 3 (ipykernel) | Idle Mode: Command Ln 1, Col 1 English (United States) 085-assignment.ipynb 26°C Haze 22:07 28-03-2023

Applied Data Science Lab | My Path Assessment Overview | work/ds-curriculum (2) - JupyterLab | +

File Edit View Run Kernel Tabs Settings Help

073-chi-square-test.ipynb x 085-assignment.ipynb x

Filter files by name

Name Last Modified

- ipymb_chec... 6 days ago
- images 7 days ago
- models 6 days ago
- .env 6 days ago
- 081-workin... 5 days ago
- 082-test-dr... 7 days ago
- 083-garch.ip... 2 months ago
- 084-model... 6 days ago
- 085-assign...** 6 days ago
- config.py 2 months ago
- data.py 7 days ago
- main.py 6 days ago
- model.py 6 days ago
- stocks.sqlite 6 days ago

Working with APIs

Task 8.5.1: Create a URL to get all the stock data for MTN Group ("MTNOY") from AlphaVantage in JSON format. Be sure to use the <https://learn-api.wqu.edu> hostname. And don't worry; your submission won't include your API key!

```
[3]: ticker = "MTNOY"
output_size = "full"
data_type = "json"

url = (
    "https://learn-api.wqu.edu/1/data-services/alpha-vantage/query?"
    "function=TIME_SERIES_DAILY&"
    "f_symbol={ticker}&"
    "f_outputsizes={output_size}&"
    "f_datatype={data_type}&"
    "f_apikey=572f37a53a7b1ffcc133ce0711c18542886ee7a2c24c697978d22dd8e5fb1c7027c9abd72d7eb3656bb9a43e6fed88e719352d1281797210639c3fe3c7481a592ada63af1a"

print("url type:", type(url))
url
```

```
[3]: url type: <class 'str'>
https://learn-api.wqu.edu/1/data-services/alpha-vantage/query?function=TIME_SERIES_DAILY&symbol=MTNOY&outputsizes=full&datatype=json&apikey=572f37a53a7b1ffcc133ce0711c18542886ee7a2c24c697978d22dd8e5fb1c7027c9abd72d7eb3656bb9a43e6fed88e719352d1281797210639c3fe3c7481a592ada63af1a3544c96ec85a32408258cf2e
c42da685163ba1714b0fa16b608c2e6fb4e82f73130250f8aa24632b0f818d1a38767310ce60ca375b7a17ec841e'
```

```
[4]: # Remove API key for submission
submission_851 = url[:170]
submission_851
```

Simple 0 2 3 5 6 Python 3 (ipykernel) | Idle Mode: Command Ln 1, Col 1 English (United States) 085-assignment.ipynb 33°C Haze 12:11 04-04-2023

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curricu (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- .ipynb_checkpoints
- images
- models
- .env
- 081-working...
- 082-test-dri...
- 083-garch.ip...
- 084-model...
- 085-assign...
- config.py
- data.py
- main.py
- model.py
- stocks.sqlite

Task 8.5.2: Create an HTTP request for the URL you created in the previous task. The grader will evaluate your work by looking at the ticker symbol in the "Meta_Data" key-value pair in your response.

```
[137]: response = requests.get(url=url)
print("response type:", type(response))
response type: <class 'requests.models.Response'>

[138]: # Get symbol in "Meta Data"
submission_852 = response.json()["Meta Data"]["2. Symbol"]
submission_852

[138]: 'MTNOY'

[139]: wget_grader.grade("Project 8 Assessment", "Task 8.5.2", submission_852)
```

Very impressive.
Score: 1

Task 8.5.3: Get status code of your `response` and assign it to the variable `response_code`.

```
[140]: response_code = response.status_code
print("code type:", type(response_code))
response_code

code type: <class 'int'>
[140]: 200
```

Simple Python 3 (ipykernel) | Idle 26°C Haze

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curricu (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- .ipynb_checkpoints
- images
- models
- .env
- 081-working...
- 082-test-dri...
- 083-garch.ip...
- 084-model...
- 085-assign...
- config.py
- data.py
- main.py
- model.py
- stocks.sqlite

Yes! Great problem solving.
Score: 1

Test-Driven Development

Task 8.5.4: Create a DataFrame `df_mtnoy` with all the stock data for MTN. Make sure that the DataFrame has the correct type of index and column names. The grader will evaluate your work by looking at the row in `df_mtnoy` for 6 December 2021.

```
[142]: response_data = response.json()
stock_data = response_data["Time Series (Daily)"]
df_mtnoy = pd.DataFrame.from_dict(stock_data, orient="index", dtype=float)
df_mtnoy.index = pd.to_datetime(df_mtnoy.index)
df_mtnoy.index.name = "date"
df_mtnoy.columns = [c.split(".")[-1] for c in df_mtnoy.columns]
print("df_mtnoy type:", type(df_mtnoy))
df_mtnoy.head()

df_mtnoy: open high low close volume
date
2023-03-27 7.01 7.0100 6.926 6.930 17262.0
2023-03-24 6.98 6.9800 6.810 6.810 9856.0
2023-03-23 7.01 7.0785 6.970 6.970 15829.0
2023-03-22 6.98 7.0500 6.950 7.030 72021.0
2023-03-21 6.59 6.6900 6.580 6.675 20334.0
```

[143]: # Get row for 6 Dec 2021

Mode: Command | Ln 8, Col 24 English (United States) 085-assignment.ipynb

26°C Haze

Applied Data Science Lab | My Path Module | WorldQuant | work/ds-curriculum (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

085-assignment.ipynb 081-working-with-apis.ipynb 083-garch.ipynb 082-test-driven.ipynb data.py model.py Python 3 (ipykernel)

Filter files by name

/ ... / ds-curriculum / 080-volatility-forecasting-in-india /

Name	Last Modified
.ipynb_checkpoints	20 minutes ago
images	2 months ago
models	2 months ago
.env	2 months ago
081-working...	8 hours ago
082-test-dri...	an hour ago
083-garch.ip...	2 months ago
084-model...	2 months ago
085-assign...	seconds ago
config.py	2 months ago
data.py	9 hours ago
main.py	2 months ago
model.py	15 minutes ago
stocks.sqlite	a minute ago

2023-03-21 6.59 6.6900 6.580 6.675 20334.0

```
[143]: # Get raw for 6 Dec 2021
submission_854 = df_mtnoy.loc["2021-12-06"].to_frame().T
submission_854

[143]:
open high low close volume
2021-12-06 10.16 10.18 10.11 10.11 13542.0

[144]: wqet_grader.grade("Project 8 Assessment", "Task 8.5.4", submission_854)

✓ Awesome work.
Score: 1
```

Task 8.5.5: Connect to the database whose name is stored in the `.env` file for this project. Be sure to set the `check_same_thread` argument to `False`. Assign the connection to the variable `connection`. The grader will evaluate your work by looking at the database location assigned to `connection`.

```
[145]: connection = sqlite3.connect(database=settings.db_name, check_same_thread=False)
connection

[145]: <sqlite3.Connection at 0x7f0ef62055d0>

[146]: # Get location of database for `connection`
submission_855 = connection.cursor().execute("PRAGMA database_list;").fetchall()[0][-1]
submission_855

[146]: '/home/jovyan/work/ds-curriculum/080-volatility-forecasting-in-india/stocks.sqlite'

[147]: wqet_grader.grade("Project 8 Assessment", "Task 8.5.5", submission_855)
```

Simple 0 5 Python 3 (ipykernel) | Idle Mode: Command Ln 8, Col 24 English (United States) 085-assignment.ipynb 26°C Haze 22:07 IN 28-03-2023

Applied Data Science Lab | My Path Module | WorldQuant | work/ds-curriculum (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

085-assignment.ipynb 081-working-with-apis.ipynb 083-garch.ipynb 082-test-driven.ipynb data.py model.py Python 3 (ipykernel)

Filter files by name

/ ... / ds-curriculum / 080-volatility-forecasting-in-india /

Name	Last Modified
.ipynb_checkpoints	20 minutes ago
images	2 months ago
models	2 months ago
.env	2 months ago
081-working...	8 hours ago
082-test-dri...	an hour ago
083-garch.ip...	2 months ago
084-model...	2 months ago
085-assign...	seconds ago
config.py	2 months ago
data.py	9 hours ago
main.py	2 months ago
model.py	15 minutes ago
stocks.sqlite	a minute ago

2023-03-21 6.59 6.6900 6.580 6.675 20334.0

```
[147]: wqet_grader.grade("Project 8 Assessment", "Task 8.5.5", submission_855)

✓ Yup. You got it.
Score: 1
```

Task 8.5.6: Insert `df_mtnoy` into your database. The grader will evaluate your work by looking at the first five rows of the `MTNOW` table in the database.

```
[148]: from data import SQLRepository
repo = SQLRepository(connections=connection)
response = repo.insert_table(table_name=ticker, records=df_mtnoy, if_exists="replace")

[149]: # Get first five rows of `MTNOW` table
submission_856 = pd.read_sql(sql="SELECT * FROM MTNOW LIMIT 5", con=connection)
submission_856

[149]:
date open high low close volume
0 2023-03-27 00:00:00 7.01 7.0100 6.926 6.930 17262.0
1 2023-03-24 00:00:00 6.98 6.9800 6.810 6.810 9856.0
2 2023-03-23 00:00:00 7.01 7.0785 6.970 6.970 15829.0
3 2023-03-22 00:00:00 6.98 7.0500 6.950 7.030 72021.0
4 2023-03-21 00:00:00 6.59 6.6900 6.580 6.675 20334.0

[150]: wqet_grader.grade("Project 8 Assessment", "Task 8.5.6", submission_856)
```

Simple 0 5 Python 3 (ipykernel) | Idle Mode: Command Ln 8, Col 24 English (United States) 085-assignment.ipynb 26°C Haze 22:07 IN 28-03-2023

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curricu (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Task 8.5.7: Read the MTHOY table from your database and assign the output to df_mtnoy_read. The grader will evaluate your work by looking at the row for 27 April 2022.

```
[151]: sql = "SELECT * FROM 'MTHOY_BSE'"
df_mtnoy_read = pd.read_sql(
    sql, connection, parse_dates=["date"], index_col="date"
)

print("df_mtnoy_read type:", type(df_mtnoy_read))
print("df_mtnoy_read shape:", df_mtnoy_read.shape)
df_mtnoy_read.head()

df_mtnoy_read type: <class 'pandas.core.frame.DataFrame'>
df_mtnoy_read shape: (3954, 5)

      open  high  low  close  volume
      date
2023-03-27  7.01  7.0100  6.926  6.930  17262.0
2023-03-24  6.98  6.9800  6.810  6.810  9856.0
2023-03-23  7.01  7.0785  6.970  6.970  15829.0
2023-03-22  6.98  7.0500  6.950  7.030  72021.0
2023-03-21  6.59  6.6900  6.580  6.675  20340.0

[152]: # Get row for 27 April 2022
submission_857 = df_mtnoy_read.loc["2022-04-27"].to_frame().T
submission_857
```

Simple Python 3 (ipykernel) Idle Mode: Command Ln 8, Col 24 English (United States) 085-assignment.ipynb

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curricu (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Predicting Volatility

Prepare Data

Task 8.5.8: Create a Series y_mtnoy with the 2,500 most recent returns for MTN. The grader will evaluate your work by looking at the volatility for 9 August 2022.

```
[155]: def wrangle_data(ticker, n_observations):
    """Extract data from database (or get from AlphaVantage), transform it
    for training model, and attach it to 'self.data'.

    Parameters
    -----
    n_observations : int
        Number of observations to retrieve from database

    Returns
    -----
    None
    """
    # Add new data to database if required
    df = repo.read_table(table_name=ticker, limit=n_observations+1)

    # Pull data from SQL database
    df.sort_index(ascending=True, inplace=True)

    # Clean data, attach to class as 'data' attribute
    df['return'] = df['close'].pct_change() * 100

    return df[['return']].dropna()
```

Simple Python 3 (ipykernel) Idle Mode: Command Ln 1, Col 1 English (United States) 085-assignment.ipynb

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curriculum (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- 085-assignment.ipynb 5 minutes ago
- config.py 2 months ago
- data.py 10 hours ago
- main.py 2 months ago
- model.py 2 months ago
- stocks.sqlite 9 minutes ago

```
[156]: y_mtney = wrangle_data(ticker = ticker, n_observations=2500)

print("y_mtney type:", type(y_mtney))
print("y_mtney shape:", y_mtney.shape)
y_mtney.head()

y_mtney type: <class 'pandas.core.series.Series'>
y_mtney shape: (2500,)

[156]: date
2013-04-23 2.305476
2013-04-24 2.760563
2013-04-25 -0.383772
2013-04-26 -1.320859
2013-04-29 3.346347

Name: return, dtype: float64

[157]: # Get data for 8 Aug 2022
submission_859 = float(y_mtney["2022-08-09"])
submission_859

[157]: 1.5783540022547893

[158]: wqet_grader.grade("Project 8 Assessment", "Task 8.5.8", submission_859)
```

Way to go!
Score: 1

Task 8.5.9: Calculate daily volatility for `y_mtney`, and assign the result to `mtney_daily_volatility`.

```
[ ]: mtney_daily_volatility = ...
```

Python 3 (pykernel) | Idle Mode: Command Ln 1, Col 1 English (United States) 085-assignment.ipynb

26°C Haze

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curriculum (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- 085-assignment.ipynb 31 minutes ago
- config.py 2 months ago
- data.py 10 hours ago
- main.py 2 months ago
- model.py 2 months ago
- stocks.sqlite 11 minutes ago

Task 8.5.9: Calculate daily volatility for `y_mtney`, and assign the result to `mtney_daily_volatility`.

```
[159]: mtney_daily_volatility = y_mtney.std()

print("mtney_daily_volatility type:", type(mtney_daily_volatility))
print("MTN Daily Volatility:", mtney_daily_volatility)

mtney_daily_volatility type: <class 'float'>
MTN Daily Volatility: 2.9155559834864886

[160]: wqet_grader.grade("Project 8 Assessment", "Task 8.5.9", mtney_daily_volatility)
```

Yes! Your hard work is paying off.
Score: 1

Task 8.5.10: Calculate the annual volatility for `y_mtney`, and assign the result to `mtney_annual_volatility`.

```
[161]: import numpy as np

[163]: mtney_annual_volatility = mtney_daily_volatility* np.sqrt(252)

print("mtney_annual_volatility type:", type(mtney_annual_volatility))
print("MTN Annual Volatility:", mtney_annual_volatility)

mtney_annual_volatility type: <class 'numpy.float64'>
MTN Annual Volatility: 46.28301639474954

[164]: wqet_grader.grade("Project 8 Assessment", "Task 8.5.10", float(mtney_annual_volatility))
```

Excellent! Keep going.

Python 3 (pykernel) | Idle Mode: Command Ln 1, Col 1 English (United States) 085-assignment.ipynb

26°C Haze

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curriculum (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- 085-assignment.ipynb 34 minutes ago
- 081-working-with-apis.ipynb 2 months ago
- 083-garch.ipynb 2 months ago
- 082-test-driven.ipynb 2 months ago
- 084-model.py 2 months ago
- 085-assignment.ipynb seconds ago
- config.py 2 months ago
- data.py 10 hours ago
- main.py 2 months ago
- model.py 5 months ago
- stocks.sqlite 15 minutes ago

```
[169]: # Create 'fig' and 'ax'
fig, ax = plt.subplots(figsize=(15,6))

# Plot 'y_mtney' on 'ax'
y_mtney.plot(ax=ax, label='daily return')

# Add axis labels
plt.xlabel('Date')
plt.ylabel('returns')
plt.title("Time series of MTNOY Returns")
# Add title

plt.legend()
# Don't delete the code below
plt.savefig("images/8-5-11.png", dpi=150)
```

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curriculum (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- 085-assignment.ipynb 40 minutes ago
- 081-working-with-apis.ipynb 2 months ago
- 083-garch.ipynb 2 months ago
- 082-test-driven.ipynb 2 months ago
- 084-model.py 2 months ago
- 085-assignment.ipynb seconds ago
- config.py 2 months ago
- data.py 10 hours ago
- main.py 2 months ago
- model.py 11 minutes ago
- stocks.sqlite 21 minutes ago

Task 8.5.12: Create an ACF plot of the squared returns for MTN. Be sure to label the x-axis "Lag [days]", the y-axis "Correlation Coefficient", and use the title "ACF of MTNOY Squared Returns".

```
[182]: from statsmodels.graphics.tsaplots import plot_acf

# Create 'fig' and 'ax'
fig, ax = plt.subplots(figsize=(15,6))

# Plot ACF of squared returns on 'ax'
plot_acf(y_mtney**2, ax=ax)

# Add axis labels and title
plt.xlabel("Lag[days]")
plt.ylabel("Correlation coefficient")
plt.title("ACF of MTNOY Squared Returns")

# Don't delete the code below
plt.savefig("images/8-5-12.png", dpi=150)
```

Screenshot of a Jupyter Notebook interface showing a PACF plot and code execution.

The notebook tab bar includes: Applied Data Science Lab, My Path Module | WorldQuant U, work/ds-curriculum (6) - JupyterLab, Introducing ChatGPT, New chat, and Python 3 (ipykernel).

The left sidebar shows a file tree for the directory `/ ... / ds-curriculum / 080-volatility-forecasting-in-india /`. The current file is `085-assignment.ipynb`.

The main area displays a partial PACF plot with the following data points:

Lag (days)	Correlation coefficient
0	~0.95
1	~0.35
2	~0.15
3	~0.20
4	~0.18
5	~0.22
6	~0.15
7	~0.18
8	~0.12
9	~0.15
10	~0.18
11	~0.12
12	~0.15
13	~0.12
14	~0.15
15	~0.12
16	~0.15
17	~0.12
18	~0.15
19	~0.12
20	~0.15
21	~0.12
22	~0.15
23	~0.12
24	~0.15
25	~0.12
26	~0.15
27	~0.12
28	~0.15
29	~0.12
30	~0.15
31	~0.12
32	~0.15
33	~0.12
34	~0.15
35	~0.12

Code cell [184] contains:

```
with open("images/8-5-12.png", "rb") as file:
    wqet_grader.grade("Project B Assessment", "Task 8.5.12", file)
```

A green checkmark icon indicates the task is impressive, with a score of 1.

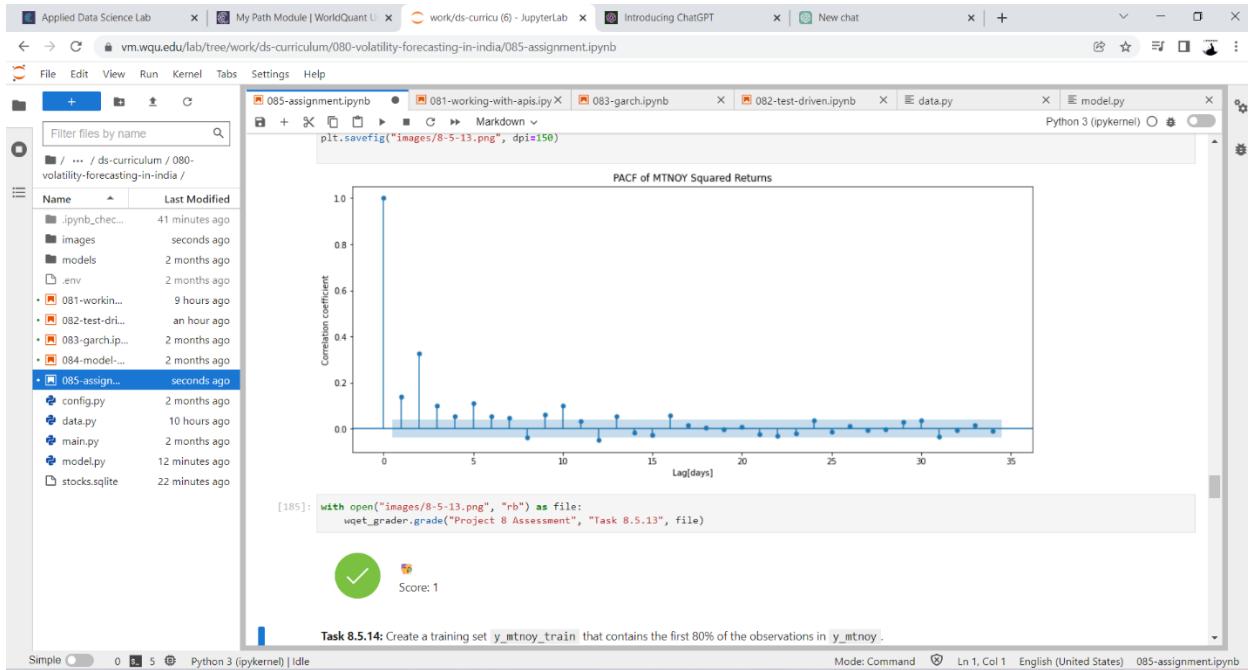
Task 8.5.13 instructions:

Task 8.5.13: Create a PACF plot of the squared returns for MTN. Be sure to label the x-axis "Lag [days]", the y-axis "Correlation Coefficient", and use the title "PACF of MTNOY Squared Returns".

Code cell [183] contains:

```
from statsmodels.graphics.tsplots import plot_pacf
```

The resulting PACF plot is titled "PACF of MTNOY Squared Returns". It shows a single data point at lag 0 with a correlation coefficient of approximately 1.0.



Task 8.5.14: Create a training set `y_mtnoy_train` that contains the first 80% of the observations in `y_mtnoy`.

```
[188]: cutoff_test = int(len(y_mtnoy)*0.8)
y_mtnoy_train = y_mtnoy[:cutoff_test]

print("y_mtnoy_train type:", type(y_mtnoy_train))
print("y_mtnoy_train shape:", y_mtnoy_train.shape)
y_mtnoy_train.head()

y_mtnoy_train type: <class 'pandas.core.series.Series'>
y_mtnoy_train shape: (2000,)

[188]: date
2013-04-23    2.305476
2013-04-24    2.760563
2013-04-25   -0.383772
2013-04-26   -1.320859
2013-04-29    3.346347
Name: return, dtype: float64

[189]: wqet_grader.grade("Project 8 Assessment", "Task 8.5.14", y_mtnoy_train)
```

Excellent! Keep going.
Score: 1

Build Model

Task 8.5.15: Build and fit a GARCH model using the data in `y_mtnoy`. Try different values for `p` and `q`, using the summary to assess its performance. The grader will evaluate whether your `model` is the correct data type.

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curriculum (6) - JupyterLab | Introducing ChatGPT | New chat | + | - | X

File Edit View Run Kernel Tabs Settings Help

Filter files by name

/ ... / ds-curriculum / 080-volatility-forecasting-in-india /

Name	Last Modified
.ipynb_checkpoints	44 minutes ago
images	4 minutes ago
models	2 months ago
.env	2 months ago
081-working...	9 hours ago
082-test-dri...	an hour ago
083-garch.ip...	2 months ago
084-model...	2 months ago
085-assign...	seconds ago
config.py	2 months ago
data.py	10 hours ago
main.py	2 months ago
model.py	15 minutes ago
stocks.sqlite	25 minutes ago

Task 8.5.15: Build and fit a GARCH model using the data in `y_mtnoy`. Try different values for `p` and `q`, using the summary to assess its performance. The grader will evaluate whether your `model` is the correct data type.

```
[191]: from arch import arch_model
# Build and train model
model = arch_model(
y_mtnoy_train, p=1, q=1, rescale=False
).fit(disp=False)

print("model type:", type(model))

# Show model summary
print(model.summary())

model type: <class 'arch.univariate.base.ARCHModelResult'>
>>> bound method ARCHModelResult.summary</summary>
Constant Mean - GARCH Model Results
=====
Dep. Variable: return R-squared: 0.000
Mean Model: Constant Mean Adj. R-squared: 0.000
Vol Model: GARCH Log-Likelihood: -4764.31
Distribution: Normal AIC: 9536.62
Method: Maximum Likelihood BIC: 9559.02
No. Observations: 2000
Date: Tue, Mar 28 2023 Df Residuals: 1999
Time: 17:01:21 Df Model: 1
Mean Model
=====
coef std err t P>|t| 95.0% Conf. Int.
-----
mu 5.3335e-03 5.480e-02 9.733e-02 0.922 [-0.102, 0.113]
Volatility Model
=====
coef std err t P>|t| 95.0% Conf. Int.
-----
omega 0.1141 5.407e-02 2.110 3.486e-02 [8.112e-03, 0.220]
```

Simple 0 5 Python 3 (ipykernel) Idle Mode: Command Ln 1, Col 1 English (United States) 085-assignment.ipynb

26°C Haze

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curriculum (6) - JupyterLab | Introducing ChatGPT | New chat | + | - | X

File Edit View Run Kernel Tabs Settings Help

Filter files by name

/ ... / ds-curriculum / 080-volatility-forecasting-in-india /

Name	Last Modified
.ipynb_checkpoints	44 minutes ago
images	4 minutes ago
models	2 months ago
.env	2 months ago
081-working...	9 hours ago
082-test-dri...	an hour ago
083-garch.ip...	2 months ago
084-model...	2 months ago
085-assign...	seconds ago
config.py	2 months ago
data.py	10 hours ago
main.py	2 months ago
model.py	15 minutes ago
stocks.sqlite	25 minutes ago

```
[191]: Mean Model: Constant Mean Adj. R-squared: 0.000
Vol Model: GARCH Log-Likelihood: -4764.31
Distribution: Normal AIC: 9536.62
Method: Maximum Likelihood BIC: 9559.02
No. Observations: 2000
Date: Tue, Mar 28 2023 Df Residuals: 1999
Time: 17:01:21 Df Model: 1
Mean Model
=====
coef std err t P>|t| 95.0% Conf. Int.
-----
mu 5.3335e-03 5.480e-02 9.733e-02 0.922 [-0.102, 0.113]
Volatility Model
=====
coef std err t P>|t| 95.0% Conf. Int.
-----
omega 0.1141 5.407e-02 2.110 3.486e-02 [8.112e-03, 0.220]
alpha[1] 0.0705 1.685e-02 4.185 2.849e-05 [3.750e-02, 0.104]
beta[1] 0.9185 1.803e-02 50.942 0.000 [ 0.883, 0.954]

Covariance estimator: robust
ARCHModelResult, id: 0x7f8ef610f400

[192]: submission_8515 = isinstance(model, ARCHModelResult)
submission_8515
```

[192]: True

```
[193]: wget_grader.grade("Project 8 Assessment", "Task 8.5.15", [submission_8515])
```

Very impressive.
Score: 1

Simple 0 5 Python 3 (ipykernel) Idle Mode: Command Ln 1, Col 1 English (United States) 085-assignment.ipynb

26°C Haze

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curriculum (6) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- 085-assignment.ipynb an hour ago
- .ipynb_checkpoints 6 minutes ago
- images 2 months ago
- models 2 months ago
- .env 2 months ago
- 081-working-with-apis.ipynb 9 hours ago
- 082-test-driven.ipynb an hour ago
- 083-garch.ipynb 2 months ago
- 084-model.ipynb 2 months ago
- 085-assignment.ipynb seconds ago
- config.py 2 months ago
- data.py 10 hours ago
- main.py 2 months ago
- model.py 18 months ago
- stocks.sqlite 27 minutes ago

```
[194]: fig, ax = plt.subplots(figsize=(15,6))

# Plot 'y_mtnoy' on 'ax'
model.std_resid.plot(ax=ax, label='Standardized Residuals')

# Add axis labels
plt.xlabel('Date')
plt.ylabel('Value')
plt.title("MTNOY GARCH Model Standardized Residuals")
# Add title

plt.legend()
# Don't delete the code below
plt.savefig("images/8-5-16.png", dpi=150)
```

MTNOY GARCH Model Standardized Residuals

Value

Mode: Command | Python 3 (pykernel) Idle | 26°C Haze | Search | 22:33 | 28-03-2023 | ENGLISH | IN | ⚡ | 1, Col 1 | English (United States) | 085-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- 085-assignment.ipynb an hour ago
- .ipynb_checkpoints 2 minutes ago
- images 2 months ago
- models 2 months ago
- .env 2 months ago
- 081-working-with-apis.ipynb 9 hours ago
- 082-test-driven.ipynb an hour ago
- 083-garch.ipynb 2 months ago
- 084-model.ipynb 2 months ago
- 085-assignment.ipynb seconds ago
- config.py 2 months ago
- data.py 10 hours ago
- main.py 2 months ago
- model.py 20 minutes ago
- stocks.sqlite 29 minutes ago

```
[196]: from statsmodels.graphics.tsaplots import plot_acf

# Create 'fig' and 'ax'
fig, ax = plt.subplots(figsize=(15,6))

# Plot ACF of squared returns on 'ax'
plot_acf(model.std_resid**2, ax=ax)

# Add axis labels and title
plt.xlabel("Lag [days]")
plt.ylabel("Correlation coefficient")
plt.title("ACF of MTNOY GARCH Model Standardized Residuals")

# Don't delete the code below
plt.savefig("images/8-5-17.png", dpi=150)
```

ACF of MTNOY GARCH Model Standardized Residuals

Correlation coefficient

Mode: Command | Python 3 (pykernel) Idle | 26°C Haze | Search | 22:36 | 28-03-2023 | ENGLISH | IN | ⚡ | 1, Col 1 | English (United States) | 085-assignment.ipynb

Screenshot saved
Click here to start sharing it

Press **G** for Xbox Game Bar

Very impressive.
Score: 1

Model Deployment

Ungraded Task: If it's not already running, start your app server.

Task 8.5.18: Change the `fit` method of your `GarchModel` class so that, when a model is done training, two more attributes are added to the object: `self.aic` with the AIC for the model, and `self.bic` with the BIC for the model. When you're done, use the cell below to check your work.

Very impressive.
Score: 1

Model Deployment

Ungraded Task: If it's not already running, start your app server.

Task 8.5.18: Change the `fit` method of your `GarchModel` class so that, when a model is done training, two more attributes are added to the object: `self.aic` with the AIC for the model, and `self.bic` with the BIC for the model. When you're done, use the cell below to check your work.

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curriculum (4) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- .ipynb_checkpoints
- images
- models
- .env
- 081-working.ipynb
- 082-test-drl.ipynb
- 083-garch.ipynb
- 084-model-deployment.ipynb
- 085-assignment.ipynb
- config.py
- data.py
- main.py
- model.py
- stocks.sqlite

Model Deployment

Ungraded Task: If it's not already running, start your app server.

Task 8.5.18: Change the `fit` method of your `GarchModel` class so that, when a model is done training, two more attributes are added to the object: `self.aic` with the AIC for the model, and `self.bic` with the BIC for the model. When you're done, use the cell below to check your work.

Tip: How can you access the AIC and BIC scores programmatically? Every `ARCHModelResult` has an `.aic` and a `.bic` attribute.

```
[243]: from arch.univariate.base import ARCHModelResult
[244]: model.aic = ARCHModelResult.aic
model.aic
[244]: <property_cached.cached_property at 0x7f8e63616160>
[247]: # Import 'build_model' function
from main import build_model

# Build model using new 'MTNOY' data
model = build_model(tickers="MTNOY", use_new_data=True)

# Wrangle 'MTNOY' returns
model.wrangle_data(n_observations=2500)

# Fit GARCH(1,1) model to data
model.fit(p1, q1)

# Does model have AIC and BIC attributes?
assert hasattr(model, "aic")
assert hasattr(model, "bic")
```

Simple Python 3 (ipykernel) | Idle Saving completed Mode: Command Ln 1, Col 1 English (United States) 085-assignment.ipynb 25°C Haze 23:42 28-03-2023

Applied Data Science Lab | My Path Module | WorldQuant U | work/ds-curriculum (4) - JupyterLab | Introducing ChatGPT | New chat

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name Last Modified

- .ipynb_checkpoints
- images
- models
- .env
- 081-working.ipynb
- 082-test-drl.ipynb
- 083-garch.ipynb
- 084-model-deployment.ipynb
- 085-assignment.ipynb
- config.py
- data.py
- main.py
- model.py
- stocks.sqlite

```
[244]: <property_cached.cached_property at 0x7f8e63616160>
[247]: # Import 'build_model' function
from main import build_model

# Build model using new 'MTNOY' data
model = build_model(tickers="MTNOY", use_new_data=True)

# Wrangle 'MTNOY' returns
model.wrangle_data(n_observations=2500)

# Fit GARCH(1,1) model to data
model.fit(p1, q1)

# Does model have AIC and BIC attributes?
assert hasattr(model, "aic")
assert hasattr(model, "bic")

[248]: # Put test results into dictionary
submission_8518 = {"has_aic": hasattr(model, "aic"), "has_bic": hasattr(model, "bic")}
submission_8518
[248]: {'has_aic': True, 'has_bic': True}

[249]: wget_grader.grade("Project 8 Assessment", "Task 8.5.18", submission_8518)
```

Good work!
Score: 1

Task 8.5.19: Change the `fit_model` function in the `main` module so that the `"message"` it returns includes the AIC and BIC scores. For example, the message should look something like this:

Applied Data Science Lab | My Path Module | WorldQuant | My Path Module | WorldQuant | work/ds-curricu (3) - JupyterLab

File Edit View Run Kernel Tabs Settings Help

085-assignment.ipynb main.py model.py Python 3 (ipykernel)

Task 8.5.19: Change the `fit_model` function in the `main` module so that the "message" it returns includes the AIC and BIC scores. For example, the message should look something like this:

```
"Trained and saved 'models/2022-10-12T23:10:06.577238_MTNQY.pkl'. Metrics: AIC 9892.184665169907, BIC 9914.588275008075."
```

When you're done, use the cell below to check your work.

```
[119]: from main import FitIn, fit_model

# Instantiate 'FitIn' object
request = FitIn(ticker="MTNOY", use_new_data=False, n_observations=2500, p=1, q=1)

# Build model and fit to data, following parameters in 'request'
fit_out = fit_model(request=request)

# Inspect 'fit_out'
fit_out
```

```
[119]: {'ticker': 'MTNOY',
       'use_new_data': False,
       'n_observations': 2500,
       'p': 1,
       'q': 1,
       'success': True,
       'message': "Trained and saved 'models/2023-03-29T06:18:53.964219_MTNOY.pkl'. Metrics: AIC 12011.626691012409, BIC 12034.922875055834."}
```

```
[120]: wget_grader.grade("Project 8 Assessment", "Task 8.5.19", fit_out)
```

Excellent! Keep going.
Score: 1

Simple 0 2 Python 3 (ipykernel) Idle 29°C Haze

Applied Data Science Lab | My Path Module | WorldQuant | work/ds-curricu (4) - JupyterLab | My Path Module | WorldQuant | Introducing ChatGPT | PredictOut Not Defined. Mode: Command Ln 1, Col 1 English (United States) 085-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

085-assignment.ipynb main.py jovyan@user-63c7ebda1089-X 084-model-deployment.ipynb model.py Python 3 (ipykernel)

Task 8.5.20: Create a `post` request to hit the "/fit" path running at "`http://localhost:8008`". You should train a GARCH(1,1) model on 2500 observations of the MTNQY data you already downloaded. Pass in your parameters as a dictionary using the `json` argument. The grader will evaluate the JSON of your `response`.

```
[158]: # URL of '/fit' path
url = "http://localhost:8008/fit"
# Data to send to path
json = [
    {
        "ticker": "MTNOY",
        "use_new_data": False,
        "n_observations": 2500,
        "p": 1,
        "q": 1
    }
]
# Response of post request
response = requests.post(url=url, json=json)

print("response type:", type(response))
print("response status code:", response.status_code)
```

```
[159]: submission_8520 = response.json()
submission_8520
```

```
[159]: {'ticker': 'MTNOY',
       'use_new_data': False,
       'n_observations': 2500,
       'p': 1,
       'q': 1,
       'success': True,
       'message': "Trained and saved 'models/2023-03-29T08:27:40.763135_MTNOY.pkl'. : AIC 12011.626691012409, BIC 12034.922875055834."}
```

```
[160]: wget_grader.grade("Project 8 Assessment", "Task 8.5.20", submission_8520)
```

Good work!
Score: 1

Simple 1 2 Python 3 (ipykernel) Idle 34°C Haze 13:58 29-03-2023

The screenshot shows a Jupyter Notebook interface with multiple tabs at the top. The active tab is '085-assignment.ipynb'. The notebook contains several cells of Python code. The first cell (cell 161) contains code to make a POST request to a local host endpoint. The second cell (cell 162) defines a variable 'submission_8521' as the response object. The third cell (cell 163) calls a function 'wget_grader.grade' with parameters 'Project 8 Assessment', 'Task 8.5.21', and 'submission_8521'. A green checkmark icon is displayed next to the message 'Yes! Your hard work is paying off.' indicating success. The status bar at the bottom shows the date '29-03-2023', time '14:00', and battery level '34°C Haze'.

```
[161]: # URL of `/predict` path
url = "http://localhost:8008/predict"
# Data to send to path
json = {
    "ticker": "MTNQY",
    "n_days": 5
}
# Response of post request
response = requests.post(url=url, json=json)

print("response type:", type(response))
print("response status code:", response.status_code)

[162]: submission_8521 = response.json()

[162]: {'ticker': 'MTNQY',
        'n_days': 5,
        'success': False,
        'forecast': {},
        'message': "build_model() missing 1 required positional argument: 'use_new_data'"}

[163]: wget_grader.grade("Project 8 Assessment", "Task 8.5.21", submission_8521)
```

Simple Mode: Command Ln 1, Col 1 English (United States) 085-assignment.ipynb

34°C Haze

14:00 29-03-2023