



modulenofterror in python pyinstaller app

[All](#)[Videos](#)[News](#)[Images](#)[Shopping](#)[More](#)[Settings](#)[Tools](#)

About 21,000 results (0.46 seconds)

When Things Go Wrong — PyInstaller 3.3.1 documentation

<https://pyinstaller.readthedocs.io/en/v3.3.1/when-things-go-wrong.html>

It may happen that when you attempt to bundle your **app** either **PyInstaller** itself, or your bundled **app**, terminates with a **Python** traceback. Then please consider ...

People also ask

How does PyInstaller work?



How do I use PyInstaller?



How do I install PyInstaller?



How do I use py2exe?

[Feedback](#)

No module named when using PyInstaller - Stack Overflow

<https://stackoverflow.com/questions/.../no-module-named-when-using-pyinstaller>

5 answers

Jan 26, 2017 - So if you import module A, **pyinstaller** sees this. But any ... There is no need to change anything in your **python** scripts. You can directly add the ...

ModuleNotFoundError with **Pyinstaller**

19 Mar 2019

exe-file created by **pyinstaller**, not find self-defined modules ...

11 Jan 2019

Django project to .exe with **Pyinstaller** - Missing modules ...

20 Apr 2018

PyInstaller runs fine but exe file errors: No module named, Failed ...

20 Oct 2017

[More results from stackoverflow.com](#)

ModuleNotFoundError: No module named 'PyInstaller' · Issue #356 ...

<https://github.com/Azure/azure-functions-python-worker/issues/356>

Mar 5, 2019 - ... my **Python** module, I get the following error **ModuleNotFoundError**: No module ...
Timestamp: Function **App** name: NA; Function name(s) (as ...

problems with pyinstaller · Issue #538 · librosa/librosa · GitHub

<https://github.com/librosa/librosa/issues/538>

Mar 29, 2017 - Ok so I did this: **pyinstaller** test.py and test.py is the file which import ... I uninstalled numpy from my **python** environment and downloaded the ...

Missing: **app** | Must include: **app**

Issues When Using auto-py-to-exe - Nitratine

<https://nitratine.net/blog/post/issues-when-using-auto-py-to-exe/>

Nov 17, 2018 - Typically though, errors brought up in **PyInstaller** are due to ... The interface uses Chromes **app** mode and a small **Python** server in the backend **ModuleNotFoundError**: No module named x / **ImportError**: No module named x.

[PDF] PyInstaller Documentation - Read the Docs

<https://readthedocs.org/projects/pyinstaller/downloads/pdf/v3.3.1/>

Apr 8, 2018 - The user can run the packaged **app** without installing a **Python** interpreter or any modules. **PyInstaller** supports **Python** 2.7 and **Python** 3.3+, and correctly bundles the major **Python** packages such as numpy, PyQt, Django, wxPython, and others.

3 Steps convert .py to .exe – TomTom – Medium

<https://medium.com/@tomtom.analytics/3-steps-convert-py-to-exe-e9ccd64a502>

Sep 4, 2018 - Install **pyinstaller** with command "pip install **pyinstaller**"; "**pyinstaller.exe**" will located in .../Scripts/; With ... **ModuleNotFoundError**: No module named 'pandas'. ... Then, type the following command on the cmd to make a standalone **application** ... Topic Modeling and Data Visualization with **Python**/Flask.



modulenothonerror in python pyinstaller app



Sep 9, 2018 - Pyinstaller bundles a Python application and all its dependencies into a single package.

Missing: modulenothonerror | Must include: modulenothonerror

How to compile python 3.7 to exe with pyinstaller? : learnpython ...

https://www.reddit.com/.../learnpython/.../how_to_compile_python_37_to_exe_with_... ▼

Oct 19, 2018 - 3 posts - 3 authors

I'm making a console based game in python and I'm trying to compile it into an exe with pyinstaller.

When I run "pyinstaller --onefile...

Error with Pyinstaller Modules - Python Forum

<https://python-forum.io/Thread-Error-with-Pyinstaller-Modules> ▼

Jan 16, 2017 - Already tried multiple re-installations of Python but still Pyinstaller ... I followed your steps and was able to build the application in the venv and ...



More results

Searches related to modulenothonerror in python pyinstaller app

pyinstaller include modules

install pyinstaller

pyinstaller multiple python files

pyinstaller matplotlib

python to exe

pyinstaller gui

pyinstaller django

pyinstaller runtime_tmpdir

India

641018, Coimbatore, Tamil Nadu - From your Internet address - Use precise location - Learn more

Help

Send feedback

Privacy

Terms

No module named when using PyInstaller

[Ask Question](#)

▲
24 ▼
★
11

I try to compile a Python project under Windows 7 using PyInstaller. The project works fine, there are no issues, however when I try to compile it the result doesn't work. Though I get no warnings during compilation there are many in the `warnmain.txt` file in the `build` directory:
[warnmain.txt](#)

I don't really understand those warnings, for example "no module named `numpy.pi`" since `numpy.pi` is no module but a number. I never tried to import `numpy.pi`. I did import `numpy` and `matplotlib` explicitly. In addition I'm using `PyQt4`. I thought the error might be related to those libraries.

However I was able to compile a simple script which uses `numpy` successfully:

```
import sys
from PyQt4 import QtGui, QtCore
import numpy as np

class MainWindow(QtGui.QMainWindow):
    def __init__(self):
        QtGui.QMainWindow.__init__(self)

        self.pb = QtGui.QPushButton(str(np.pi), self)

app = QtGui.QApplication(sys.argv)
main = MainWindow()
main.show()
sys.exit(app.exec_())
```

Successfully here means that the created executable file actually showed the desired output. However there is also a `warnmain.txt` file created which contains exactly the same 'warnings' as the one before. So I guess the fact that compiling my actual project does not give any success is not (or at least not only) related to those warnings. But what

else could be the error then? The only output during compilation are 'INFO's and none of the is a negative statement.

I did not specify an additional hook directory but the hooks where down using the default directory as far as I could read from the compile output, e.g. hook-matplotlib was executed. I could not see any hook for numpy neither could I for my small example script but this one worked. I used the following imports in my files (not all in the same but in different ones):

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.backends.backend_qt4agg import FigureCanvasQTAgg as FigureCanvas
from matplotlib.backends.backend_qt4agg import NavigationToolbar2QTAgg as
NavigationToolbar
from PyQt4 import QtGui, QtCore
import json
import sys
import numpy # added this one later
import matplotlib # added this one later
```

Since PyInstaller does not give any errors/warnings I could not figure out if the problem is related to the libraries or if there is something else to be considered.

python numpy matplotlib pyqt4
pyinstaller

edited Nov 1 '18 at 22:29



[BartoszKP](#)

27.3k 10 73 109

asked Sep 8 '14 at 21:34





[a_guest](#)

7,858 3 15 46

do you get an error when you try to launch your program from the command line? or it just wont start and nothing is printed (does qt redirect stdout to some other place like an output window that gets closed when you crash?)... to further my comment I suspect you have a crash (probably from something that uses json/xml resources or something (not import xxxx) ... and the error message is being printed to some QT window that ends up closing when you crash ... see if qt.App has a redirect=False option or something to keep the stdout – [Joran Beasley](#)

Sep 8 '14 at 22:02 

▲ Interestingly, if I were to do something like `import numpy.sinh` I get `ImportError: No module named sinh` and it is only when I do `import numpy.numpy.sinh` that I get `ImportError: No module named numpy.sinh`. Have you checked [PyInstaller finding out what went wrong?](#) to grab errors / verbose imports / debug info? – [ssnobody](#) Sep 8 '14 at 22:05 

1 ▲ @JoranBeasley: I just tried to launch the exe from the cmd without any result, but for compiling I used the flag `-w` (see [pythonhosted.org/PyInstaller/#options-for-the-executable-output](#)) which prevents Windows from launching a console for any input/output. If I'd compile without `-w` I cannot get the graphical user interface. However the program works fine when invoking from the interpreter, so I don't think it's an error inside. @ssnobody: I'm aware of that link but it didn't bring any benefits for me. `import numpy.sinh` won't work since `sinh` is no module. – [a_guest](#) Sep 8 '14 at 22:23 

▲ @JoranBeasley: I have to correct myself, it seems to be possible to compile without `-w` and get the GUI. However if I do I get a console where I can see for a very short moment (far to short to actually read) that something's written there, but then the application terminates. The GUI does not appear. I try to get this stdout somehow to a file so I can actually read it. – [a_guest](#) Sep 8 '14 at 22:36

1 ▲ @JoranBeasley Thanks man, your suggestion was just the right way! There were actually some libraries missing (which I fixed by just importing them, see my edited question). In case you're not posting an answer I'll just summarize it in a short one. Thanks! – [a_guest](#) Sep 8 '14 at 23:16

5 Answers



Had a similar problem with `no module named FileDialog`. Discovered that

12

with version 3.2, I could use

```
pyinstaller --hidden-import
FileDialog ...
```

instead of modifying my main script.

answered Nov 8 '16 at 10:03



[Hhest](#)

129 1 6

9

Pyinstaller won't see **second level imports**. So if you import module *A*, pyinstaller sees this. But any additional module that is imported in *A* will not be seen.

There is no need to change anything in your python scripts. You can directly add the missing imports to the **spec file**. Just change the following line:

```
hiddenimports=[],
```

to

```
hiddenimports=["Tkinter", "FileDia
```

answered Jan 25 '17 at 21:44



[user1251007](#)

7,838 10 40 68

6

The problem were some runtime dependencies of matplotlib. So the compiling was fine while running the program threw some errors. Because the terminal closed itself immediately I didn't realize that. After redirecting `stdout` and `stderr` to a file I could see that I missed the libraries `Tkinter` and `FileDialog`. Adding two `import s` at the top of the main solved this problem.

answered Sep 15 '14 at 12:25

[a_guest](#)



7,858 3 15 46

- 1 ▲ This is the redirect command I used: myprogram.exe 1> errors.txt 2>&1 --- I got the same result as you... I was missing Tkinter andFileDialog modules as well :) Thanks, much appreciated. – [panofish](#) Jul 13 '15 at 14:04 ✎
- ▲ So is the problem that PyInstaller doesn't pick up second level imports? Or was this not a project with a setup.py? – [Efren](#) Mar 3 '16 at 3:25
- ▲ Yes. If you import a module and this module imports other modules itself then PyInstaller apparently won't realize those second level imports. – [a_guest](#) Mar 4 '16 at 9:58

- 1 ▲ If the matter is that you don't need Tkinter and friends because you are using PyQt4, then it might be best to avoid loading Tkinter etc altogether. ▼ Look into /etc/matplotlibrc and change the defaults to PyQt4, see the 'modified' lines below:

```
#### CONFIGURATION BEGINS HERE

# The default backend; one of GTK
# CocoaAgg MacOSX Qt4Agg Qt5Agg Tk
# Template.
# You can also deploy your own bac
# referring to the module name (wh
# 'module://my_backend'.

#modified
#backend      : TkAgg
backend       : Qt4Agg

# If you are using the Qt4Agg back
# to use the PyQt4 bindings or the
# the underlying Qt4 toolkit.

#modified
#backend.qt4 : PyQt4          # PyQt
backend.qt4  : PyQt4         # PyQt4
```

answered Apr 7 '17 at 17:26

[ullix](#)
43 1 10

- 0 ▲ I had the same problem with pyinstaller 3.0 and weblib . Importing it in the main didn't help.



Upgrading to 3.1 and deleting all build files helped.

```
pip install --upgrade pyinstaller
```

answered Jan 12 '16 at 21:06



fivef

1,236 11 13