

PMI - Engines unavailable (Python...



[Ingo Levin](#) 10 posts since Apr 12, 2018

PMI - Engines unavailable (Python Scikit-Learn) Apr 12, 2018 3:07 PM

Hi, I am on Windows8.1 and have Miniconda3 installed with Python 3.6.

I can verify this on the command line.

```
C:\Program Files\pdi-ce-8.0.0.0-28\data-integration>python -c "import sys; print (sys.executable);"
>> C:\Users\ingo\Miniconda3\python.exe
```

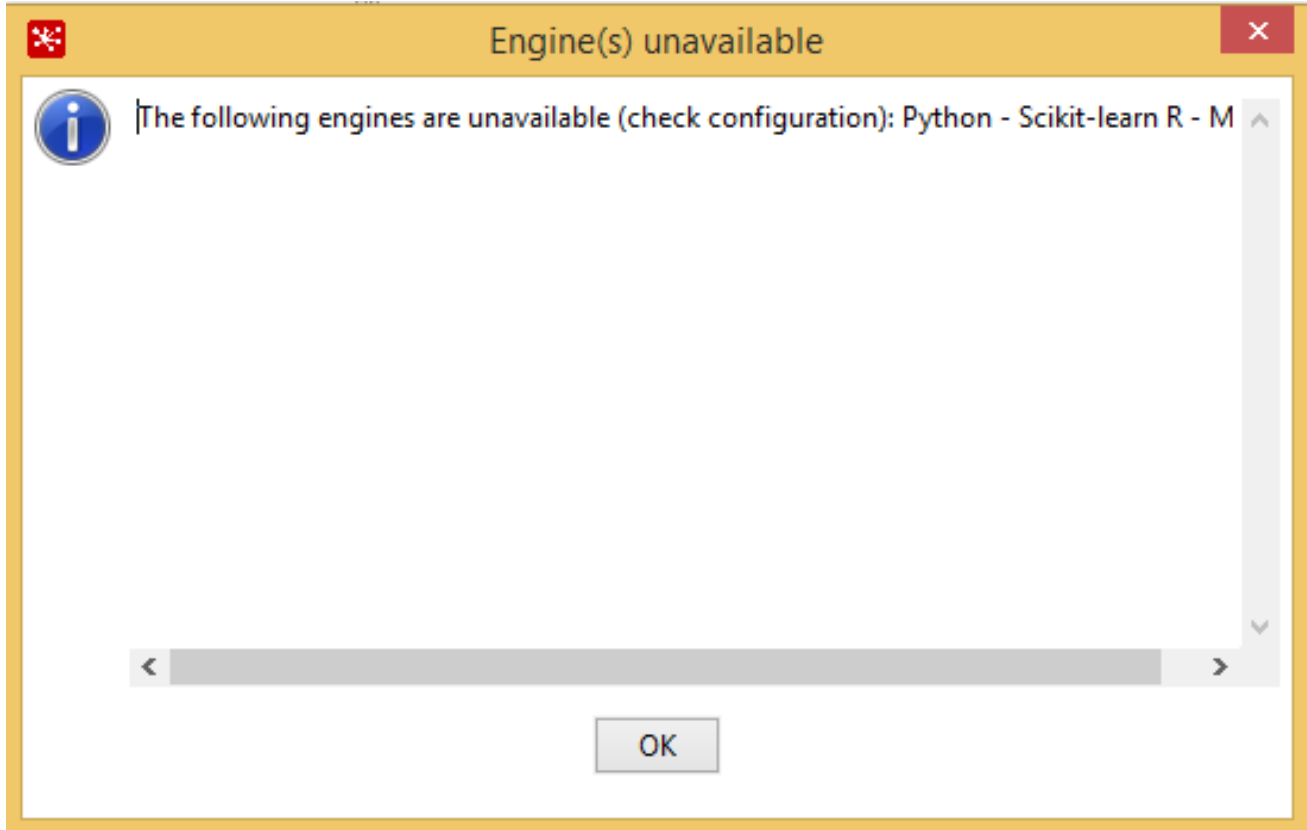
```
C:\Program Files\pdi-ce-8.0.0.0-28\data-integration>python -V
>>Python 3.6.0 :: Continuum Analytics, Inc.
```

In it I have numpy, sklearn, pandas and matplotlib installed. They are on the list of returned modules when I run:

```
python -c "help('modules')"
```

Yet, when I start PDI and try to use a PMI step, it throws an Error that Python Scikit-learn and R are not installed.

What am I doing wrong??



PMI - Engines unavailable (Python...

Tags: python, pmi



David Huh 1 posts since Sep 20, 2017

Re: PMI - Engines unavailable (Python Scikit-Learn) Apr 12, 2018 5:23 PM

Can you try installing R packages as well?

Link to install doc for your reference: https://community.hitachivantara.com/servlet/JiveServlet/downloadBody/1010952-102-1-293961/PMI_Installation__Windows.pdf

Link to other references: "PMI Installation, Developer Guide and Sample/Demos"



Ingo Levin 10 posts since Apr 12, 2018

Re: PMI - Engines unavailable (Python Scikit-Learn) Apr 17, 2018 11:26 AM

No yet. Was about to try that when I ran out of time.

I think the issue with python is that pandas does not get imported correctly - see [my comment](#) to Mark Hall.

Trying to get that sorted out first, then will get back onto R.



Ken Wood 59 posts since Jun 12, 2013

Re: PMI - Engines unavailable (Python Scikit-Learn) Apr 12, 2018 10:50 PM

Can you provide your environmental variables that you've defined for PMI? Also, how are you launching "spoon"?



Ingo Levin 10 posts since Apr 12, 2018

Re: PMI - Engines unavailable (Python Scikit-Learn) Apr 17, 2018 11:22 AM

Hi **Ken Wood** , I haven't defined any specific env variables for PMI. I am not (yet) using R, so per the docs all I should need is my python executable available in the system path. which it is.

```
C:\Users\ingo>echo %PATH%
```

```
C:\Program Files\Microsoft MPI\Bin\;C:\Program Files\PHP\v7.0;C:\Program Files (x86)\Intel\CLS Client\;C:\Program Files\Intel\CLS Client\;C:\ProgramData\Oracle\Java\javapath;C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Program Files (x86)\Intel\Intel(R) Management Engine Components\DAL;C:\Program Files\Intel\Intel(R) Management Engine Components\DAL;C:\Program Files (x86)\Intel\Intel(R) Management Engine Components\IPT;C:\Program Files\Intel\Intel(R) Management Engine Components\IPT;C:\Program Files (x86)\PuTTY\;C:\Program Files (x86)\Gartle\SaveToDB\;;C:\Android;C:\Users\ingo\Miniconda3;C:\Users\ingo\Miniconda3\Scripts;C:\Users\ingo\Miniconda3\Library\bin;C:\adb;C:\Program Files (x86)\Skype\Phone\;C:\Program Files\Microsoft SQL Server\Client SDK\ODBC\130\Tools\Binn\;C:\Program Files (x86)\Microsoft SQL Server\140\Tools\Binn\;C:\Program Files\Microsoft SQL Server\140\Tools\Binn\;C:\Program Files\Microsoft SQL Server\140\DTS\Binn\;C:\Program Files (x86)\Microsoft SQL
```

PMI - Engines unavailable (Python...

Server\Client SDK\ODBC\130\Tools\Binn\;C:\Program Files (x86)\Microsoft SQL Server\140\DTs\Binn\;C:\Program Files (x86)\Microsoft SQL Server\140\Tools\Binn\ManagementStudio\;C:\Users\ingo\Miniconda3\Scripts;C:\Users\ingo\Miniconda3\Library\bin

I am starting spoon via the shipped Spoon.bat. It's my local laptop, I'm just using the defaults and not setting explicit _PENTAHO_JAVA_HOME, KETTLE_HOME, etc.

Should I?



[Mark Hall](#) 14 posts since May 13, 2016

Re: PMI - Engines unavailable (Python Scikit-Learn) Apr 12, 2018 11:07 PM

Do you have scipy installed? PMI also requires this. You can run

```
python ~/wekafiles/packages/wekaPython/resources/py/pyCheck.py
```

(Adapt the above for Windows with respect to pointing to your home directory and backslashes etc.).

This is what it checks for:

```
def check_libraries():
    check_min_python()
    isPython3 = sys.version_info >= (3, 0)
    if isPython3:
        check_library('io')
    else:
        check_library('StringIO')
    check_library('math')
    check_library('traceback')
    check_library('socket')
    check_library('struct')
    check_library('os')
    check_library('json')
    check_library('base64')
    check_library('pickle')
    check_library('scipy')
    check_library('sklearn')
    check_library('matplotlib')

    check_library('numpy')
```

PMI - Engines unavailable (Python...

If there is no output, then you should have all the required python libraries.

Cheers,

Mark.



[Ingo Levin](#) 10 posts since Apr 12, 2018

Re: PMI - Engines unavailable (Python Scikit-Learn) Apr 17, 2018 11:28 AM

Hi Mark,

the script returns the following:

```
C:\Users\ingo>python wekafiles\packages\wekaPython\resources\py\pyCheck.py
>>>A problem occurred when trying to import pandas
```

So, this clearly seems to be the cause of my problem.

But the thing is, I do have scipy and pandas correctly installed and I can import pandas without problems in an interactive python shell session for example...

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\ingo>python wekafiles\packages\wekaPython\resources\py\pyCheck.py
A problem occurred when trying to import pandas

C:\Users\ingo>python
Python 3.6.0 [Continuum Analytics, Inc.] (default, Dec 23 2016, 11:57:41) [MSC v.
.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import pandas as pd
>>>
```

How can I troubleshoot this?

I haven't set any specific env variables for a specific python home. It's just the one referenced in the system-wide %PATH%. I have a few more other conda (python) environments, but that shouldn't matter as they are not in the PATH

My Full list of installed modules:

>>help ("modules")

Crypto	brain_stdlib	mmsystem	sspi
IPython	builtins	modulefinder	sspicon
OleFileIO_PL	bz2	mpl_toolkits	stat
OpenSSL	cProfile	msilib	statistics
PIL	calendar	msvcrt	storemagic
PyQt5	certifi	multiprocessing	string
TBB	cffi	nbconvert	stringprep
__future__	cgi	nbformat	struct
_ast	cgilib	netbios	subprocess
_asyncio	chardet	netrc	sunau
_bisect	chunk	nntplib	symbol
_blake2	clyent	notebook	sympyprinting
_bootlocale	cmath	nt	symtable
_bz2	cmd	ntpath	sys
_cffi_backend	code	ntsecuritycon	sysconfig
_codecs	codecs	nturl2path	tabnanny
_codecs_cn	codeop	numbers	tarfile
_codecs_hk	collections	numexpr	tbb
_codecs_iso2022	colorama	numpy	telnetlib
_codecs_jp	coloursys	numpydoc	tempfile
_codecs_kr	commctrl	odbc	test
_codecs_tw	compileall	olefile	test_path
_collections	concurrent	opcode	test_pycosat
_collections_abc	conda	operator	testpath
_compat_pickle	conda_env	optparse	tests
_compression	configparser	os	textwrap
_csv	contextlib	pandas	this
_ctypes	copy	pandocfilters	threading
_ctypes_test	copyreg	parser	time
_datetime	crypt	path	timeit
_decimal	cryptography	pathlib	timer
_dummy_thread	csv	pdb	tkinter
_elementtree	ctypes	pep8	token
_functools	curses	perfmon	tokenize
_hashlib	cwp	pickle	tornado
_heapq	cycler	pickleshare	trace
_imp	cythonmagic	pickletools	traceback
_io	daal	pip	tracemalloc
_json	datetime	pipes	traitlets
_license	dateutil	pkg_resources	tty

_locale	dbi	pkgutil	turtle
_lsprof	dbm	platform	turtledemo
_lzma	dde	plistlib	types
_markupbase	decimal	poplib	typing
_md5	decorator	posixpath	unicodedata
_msi	difflib	pprint	unittest
_multibytecodec	dis	profile	untitled0
_multiprocessing	distutils	prompt_toolkit	urllib
_nsis	doctest	pstats	urllib3
_opcode	docutils	psutil	uu
_operator	dummy_threading	pty	uuid
_osx_support	easy_install	py_compile	venv
_overlapped	email	pyasn1	warnings
_pickle	encodings	pyclbr	wave
_pydecimal	ensurepip	pycosat	wcwidth
_pyio	entrypoints	pycparser	weakref
_random	enum	pydoc	webbrowser
_sha1	errno	pydoc_data	wheel
_sha256	faulthandler	pyexpat	widetsnbextension
_sha3	filecmp	pyflakes	win2kras
_sha512	fileinput	pygments	win32api
_signal	fnmatch	pylab	win32clipboard
_sitebuiltins	formatter	pylint	win32com
_socket	fractions	pyparsing	win32con
_sqlite3	ftplib	pythoncom	win32console
_sre	functools	pytz	win32cred
_ssl	gc	pywin	win32crypt
_stat	genericpath	pywin32_testutil	win32cryptcon
_string	getopt	pywintypes	win32event
_strptime	getpass	qtawesome	win32evtlog
_struct	gettext	qtconsole	win32evtlogutil
_symtable	glob	qtpy	win32file
_system_path	gzip	queue	win32gui
_testbuffer	hashlib	quopri	win32gui_struct
_testcapi	heapq	random	win32help
_testconsole	hmac	rasutil	win32inet
_testimportmultiple	html	re	win32inetcon
_testmultiphase	html5lib	regcheck	win32job
_thread	http	regutil	win32lz
_threading_local	idlelib	reprlib	win32net
_tkinter	idna	requests	win32netcon
_tracemalloc	imagesize	rlcompleter	win32pdh
_warnings	imaplib	rmagic	win32pdhquery

_weakref	imghdr	rope	win32pdhutil
_weakrefset	imp	ruamel_yaml	win32pipe
_win32sysloader	importlib	run	win32print
_winapi	inspect	runpy	win32process
_winxptheme	io	sched	win32profile
abc	ipaddress	scipy	win32ras
adodbapi	ipykernel	secrets	win32rcparser
afxres	ipython_genutils	select	win32security
aifc	ipywidgets	selectors	win32service
alabaster	isapi	servicemanager	win32serviceutil
anaconda_navigator	isort	setuptools	win32timezone
antigravity	itertools	shelve	win32trace
argparse	jedi	shlex	win32traceutil
array	jinja2	shutil	win32transaction
asn1crypto	json	signal	win32ts
ast	jsonschema	simplegeneric	win32ui
astroid	jupyter	sip	win32uiOLE
asynchat	jupyter_client	sipconfig	win32verstamp
asyncio	jupyter_console	sipdistutils	win32wnet
asyncore	jupyter_core	site	win_inet_pton
atexit	keyword	six	win_unicode_console
audioop	lazy_object_proxy	sklearn	winerror
autoreload	lib2to3	smtpd	winioctlcon
babel	linecache	smtpplib	winnt
backports	locale	sndhdr	winperf
base64	logging	snowballstemmer	winreg
bdb	lzma	socket	winsound
binascii	macpath	socketserver	winxpgui
binhex	macurl2path	socks	winxptheme
binstar_client	mailbox	sockshandler	wrapt
bisect	mailcap	sphinx	wsgiref
bleach	markupsafe	spyder	xdrlib
brain_builtin_inference	marshal	spyder_breakpoints	xml
brain_dateutil	math	spyder_io_dcm	xmlrpc
brain_gi	matplotlib	spyder_io_hdf5	xxsubtype
brain_mechanize	menuinst	spyder_profiler	yaml
brain_nose	mimetypes	spyder_pylint	zipapp
brain_numpy	mistune	sqlite3	zipfile
brain_pytest	mkl_fft	sre_compile	zipimport
brain_qt	mkl_random	sre_constants	zlib
brain_six	mmap	sre_parse	zmq
brain_ssl	mmapfile	ssl	

Enter any module name to get more help. Or, type "modules spam" to search for modules whose name or summary contain the string "spam".



[Ingo Levin](#) 10 posts since Apr 12, 2018

Re: PMI - Engines unavailable (Python Scikit-Learn) May 7, 2018 8:19 AM

[Mark Hall](#)

I found out why the pyCheck script is throwing the error 'A problem occurred when trying to import pandas'.

My installed pandas version is 0.22.0+0.ga00154d.dirty which is later than the required min version 0.7.0
FYI - I am using the Intel Python distro for the faster performance, not the standard Anaconda channel.

The problem is that the pyCheck script is essentially comparing the `len(__version__.split('.'))` of both version strings and if they don't have the same length, it will immediately throw said error.

```
len('0.22.0+0.ga00154d.dirty'.split('.'))
```

```
Out[13]: 5
```

```
while
```

```
len('0.7.0'.split('.'))
```

```
Out[14]: 3
```

I think this is a bug in the pyCheck script.

I have 0.22.x installed which satisfies the min requirement 0.7.0, so there should not be an error.

SOLUTION:

I commented out these two rows in the pyCheck.py script. Now the python scikit-learn engine is available when I start PDI.

```
def check_min_pandas():
    min_pandas = pandas_version_min.split('.')
    try:
        import pandas

        actual_pandas = pandas.__version__.split('.')
    #     if len(actual_pandas) is not len(min_pandas):
    #         raise Exception()
    result = check_min_version(min_pandas, actual_pandas)
    if result:
```


PMI - Engines unavailable (Python...

```
    append_to_results(  
        'Installed pandas does not meet the minimum requirement: version ' + pandas_version_min)  
except:  
    append_to_results('A problem occurred when trying to import pandas')
```



[Mark Hall](#) 14 posts since May 13, 2016

Re: PMI - Engines unavailable (Python Scikit-Learn) May 8, 2018 3:25 AM

Cool! Thanks Ingo. I'll incorporate this fix into the next release of the wekaPython package and the PDI CPython script executor step.

Cheers,
Mark.