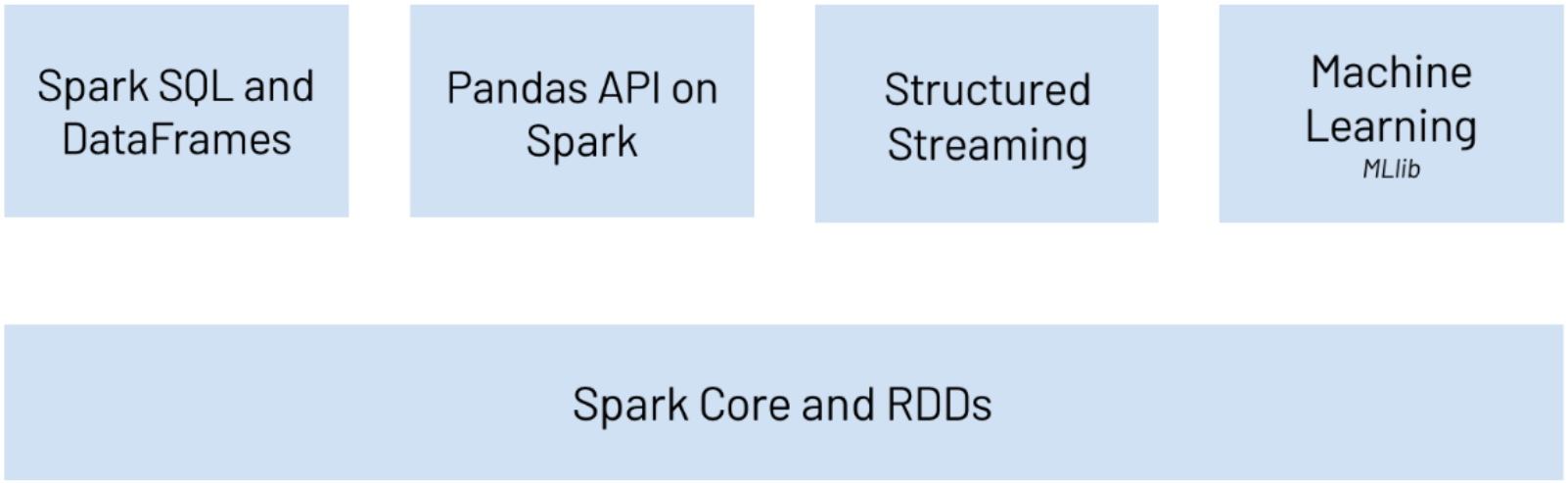
Lab2_PySpark

PySpark


Spark SQL and DataFrames
Pandas API on Spark
Structured Streaming
Machine Learning
MLlib
Spark Core and RDDs

spark.apache.org

https://spark.apache.org/docs/latest/api/python/index.html

https://spark.apache.org/docs/latest/api/python/getting_started/index.html

Installation

Using Conda

```
conda create -n pyspark_env
conda activate pyspark_env

conda install -c conda-forge pyspark


conda install plotly
conda install -c conda-forge cufflinks-py
```

DataFrame

[1]:
```
from pyspark.sql import SparkSession

spark = SparkSession.builder.getOrCreate()
```

[2]:
```
from datetime import datetime, date
import pandas as pd
from pyspark.sql import Row

df = spark.createDataFrame([
    Row(a=1, b=2., c='string1', d=date(2000, 1, 1), e=datetime(2000, 1, 1, 12, 0)),
    Row(a=2, b=3., c='string2', d=date(2000, 2, 1), e=datetime(2000, 1, 2, 12, 0)),
    Row(a=4, b=5., c='string3', d=date(2000, 3, 1), e=datetime(2000, 1, 3, 12, 0))
])
df
```

[6]:
```
# All DataFrames above result same.
df.show()
df.printSchema()
```

и так далее...

Pandas API on Spark

[1]:
```
import pandas as pd
import numpy as np
from pyspark.sql import SparkSession
import os
os.environ["PYARROW_IGNORE_TIMEZONE"] = "1"
```

Вставляем, чтобы не было предупреждений

[3]:
```
import pyspark.pandas as ps
```

[2]:
```
s = ps.Series([1, 3, 5, np.nan, 6, 8])
```

[4]:
```
psdf = ps.DataFrame(
    {'a': [1, 2, 3, 4, 5, 6],
     'b': [100, 200, 300, 400, 500, 600],
     'c': ["one", "two", "three", "four", "five", "six"]},
    index=[10, 20, 30, 40, 50, 60])
```

[8]:
```
pdf = pd.DataFrame(np.random.randn(6, 4), index=dates, columns=list('ABCD'))
```

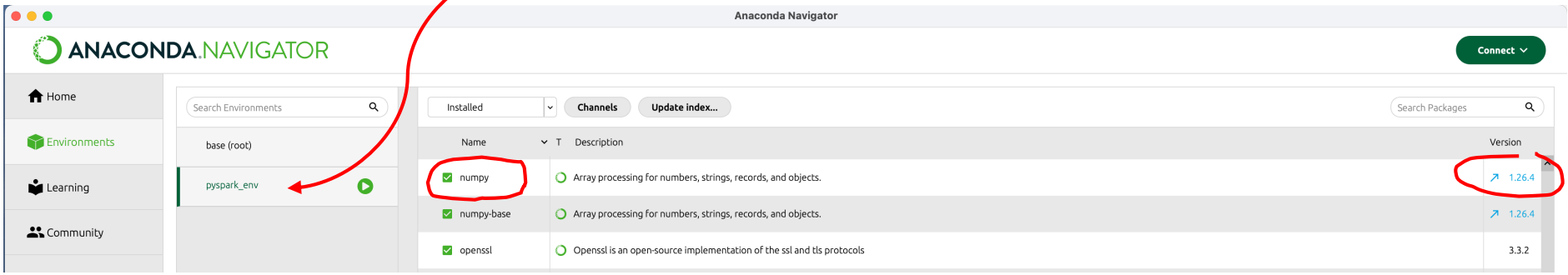
и так далее...

Ошибка при импорте pyspark.pandas, скорее всего, возникает из-за несовместимости версии NumPy с PySpark.

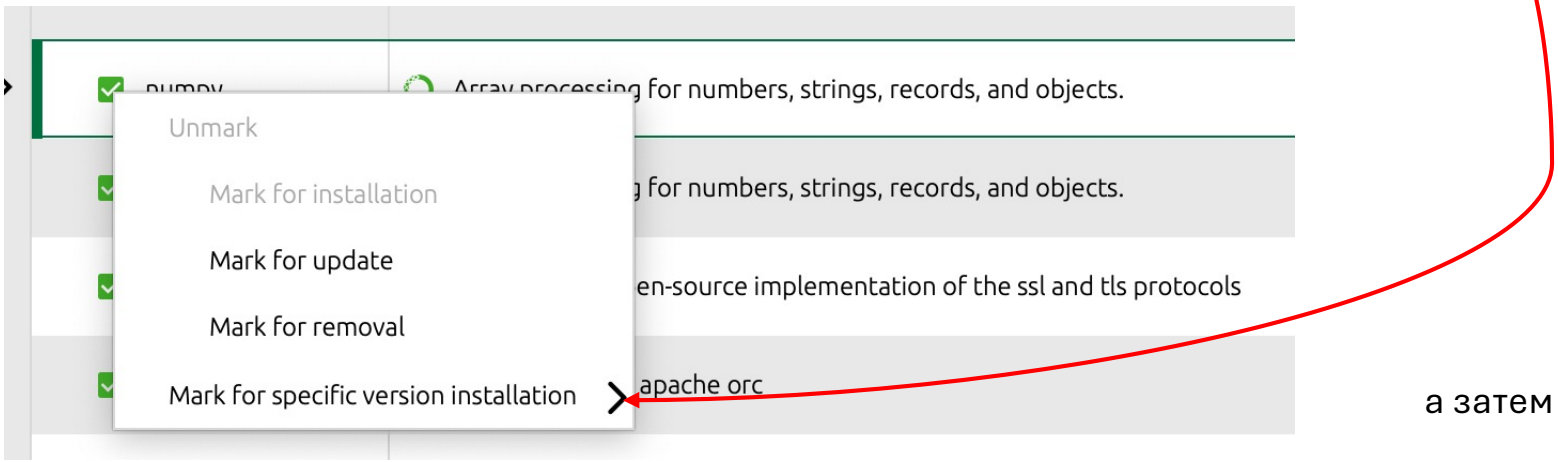
Внимание! PySpark станет совместим с NumPy 2 начиная с PySpark 4,
таким образом, версия NumPy, используемая для PySpark 3.5, ограничена и должна быть < 2.0.0.

Предупреждение: *обновление всех пакетов перед установкой PySpark может обновить NumPy до v2.1.1*

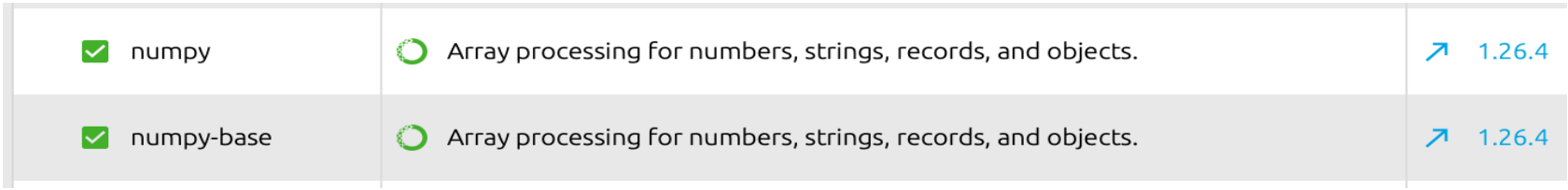
Проверяем версию NumPy в среде pyspark_env. Если больше или равна 2.0, то...



жмём правой клавишей мышки на ✅ у numpy, и во всплывающем меню выбираем



а затем версию < 2.



Spark Configurations

[34]:
```
prev = spark.conf.get("spark.sql.execution.arrow.pyspark.enabled")  # Keep its default va
ps.set_option("compute.default_index_type", "distributed")  # Use default index prevent o
import warnings
warnings.filterwarnings("ignore")  # Ignore warnings coming from Arrow optimizations.
```

[35]:
```
spark.conf.set("spark.sql.execution.arrow.pyspark.enabled", True)
%timeit ps.range(300000).to_pandas()
```

```
900 ms ± 186 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
```

[36]:
```
spark.conf.set("spark.sql.execution.arrow.pyspark.enabled", False)
%timeit ps.range(300000).to_pandas()
```

```
3.08 s ± 227 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
```

[37]:
```
ps.reset_option("compute.default_index_type")
spark.conf.set("spark.sql.execution.arrow.pyspark.enabled", prev)  # Set its default valu
```

Plotting

[42]:
```
pser = pd.Series(np.random.randn(1000),
                 index=pd.date_range('1/1/2000', periods=1000))
```

[43]:
```
psser = ps.Series(pser)
```

[44]:
```
psser = psser.cummax()
```

[45]:
```
psser.plot()
```