Base image configuration

Databricks allows for the execution of custom docker containers when spinning up clusters. This holds some advantages for certain scenarios, none of which apply at this exact moment in the EAC Azure cloud journey.

According to the documentation these are some of the scenarios. Breaking these down, we can give some reasoning here as to why batch processing doesn't require docker to run in production.

- Library customization: you have full control over the system libraries you want installed.

  Control is currently implemented via the databricks cli. https://docs.databricks.com/dev-tools/cli/libraries-cli.html If you examine the MLOps architecture, you'll not have direct access to the databricks cluster that runs in production as a model developer. How will you control your libraries in that cluster? When the workspace is instantiated you will actuate on the workspace via CICD. That CICD routine will take environment.yml files for the model and attempt to load the versions.

- Golden container environment: your Docker image is a locked down environment that will never change.

  Model developers do not have direct access to the databricks workspace which will execute the final jobs of scoring with a model.

- Docker CI/CD integration: you can integrate Azure Databricks with your Docker CI/CD pipelines.

  This scenario makes sense if you have some pre-existing Non-Databricks workflows that you want to support in Azure Databricks. This is not the case, as GBI had chosen not to continue with previous architectures for model management and instead gone with a pure databricks approach from end to end.

In the event this topic gets revisted, the details below could become the basis for a base image utilized in databricks clusters. One advantage hinted at in literature online would be on speed of spinning up clusters. https://slacker.ro/2020/12/09/how-retina-uses-databricks-container-services-to-improve-efficiency-and-reduce-costs/ But again the primary focus right now is to ensure full compliant functionality can replace prior architectures.

This page contains the detail about the configuration of the base image that will be used for packaging the models as containers.

How to build a custom base image?

To build a custom base image please follow this https://docs.microsoft.com/en-us/azure/databricks/clusters/custom-containers#step-1-build-your-base for more details.

There is https://github.com/databricks/containersthat can be referenced.

List of packages

The list of packages and their version should match based on the runtime that is selected. The details about that can be found here. *Do we need to build a custom image for each runtime version?*

To start with, we can use runtime 7.6ML libraries.

python libraries

```
name: databricks-ml
channels:
  - <Jfrog conda channels>
dependencies:
  - _libgcc_mutex=0.1=main
  - absl-py=0.9.0=py37_0
  - asn1crypto=1.3.0=py37_1
  - astor=0.8.0=py37_0
  - backcall=0.1.0=py37_0
  - backports=1.0=pyhd3eb1b0_2
  - bcrypt=3.2.0=py37h7b6447c_0
  - blas=1.0=mkl
  - blinker=1.4=py37_0
  - boto3=1.12.0=py_0
  - botocore=1.15.0=py_0
  - c-ares=1.17.1=h27cfd23_0
  - ca-certificates=2021.1.19=h06a4308_1 # (updated from h06a4308_0 in May 18, 2021 maintenance update)
  - cachetools=4.2.0=pyhd3eb1b0_0
  - certifi=2020.12.5=py37h06a4308_0
  - cffi=1.14.0=py37he30daa8_1 # (updated from py37h2e261b9_0 in May 18, 2021 maintenance update)
  - chardet=3.0.4=py37h06a4308_1003
  - click=7.0=py37_0
  - cloudpickle=1.4.1=py_0
  - configparser=3.7.4=py37_0
  - cpuonly=1.0=0
  - cryptography=2.8=py37h1ba5d50_0
  - cycler=0.10.0=py37_0
  - cython=0.29.15=py37he6710b0_0
  - decorator=4.4.1=py_0
  - dill=0.3.1.1=py37_1
  - docutils=0.15.2=py37_0
  - entrypoints=0.3=py37_0
  - flask=1.1.1=py_1
  - freetype=2.9.1=h8a8886c_1
  - future=0.18.2=py37_1
  - gast=0.3.3=py_0
  - gitdb=4.0.5=py_0
  - gitpython=3.1.0=py_0
  - google-auth=1.11.2=py_0
  - google-auth-oauthlib=0.4.1=py_2
  - google-pasta=0.2.0=py_0
  - grpcio=1.27.2=py37hf8bcb03_0
  - gunicorn=20.0.4=py37_0
  - h5py=2.10.0=py37h7918eee_0
  - hdf5=1.10.4=hb1b8bf9_0
  - icu=58.2=he6710b0_3
  - idna=2.8=py37_0
  - intel-openmp=2020.0=166
  - ipykernel=5.1.4=py37h39e3cac_0
  - ipython=7.12.0=py37h5ca1d4c_0
  - ipython_genutils=0.2.0=pyhd3eb1b0_1
  - isodate=0.6.0=py_1
  - itsdangerous=1.1.0=py37_0
  - jedi=0.17.2=py37h06a4308_1
  - jinja2=2.11.1=py_0
  - jmespath=0.10.0=py_0
  - joblib=0.14.1=py_0
  - jpeg=9b=h024ee3a_2
  - jupyter_client=5.3.4=py37_0
  - jupyter_core=4.6.1=py37_0
  - kiwisolver=1.1.0=py37he6710b0_0
  - krb5=1.17.1=h173b8e3_0 # (updated from 1.16.4 in May 18, 2021 maintenance update)
  - ld_impl_linux-64=2.33.1=h53a641e_7
  - libedit=3.1.20181209=hc058e9b_0
  - libffi=3.3=he6710b0_2 # (updated from 3.2.1 in May 18, 2021 maintenance update)
  - libgcc-ng=9.1.0=hdf63c60_0
  - libgfortran-ng=7.3.0=hdf63c60_0
  - libpng=1.6.37=hbc83047_0
  - libpq=12.2=h20c2e04_0 # (updated from 11.2 in May 18, 2021 maintenance update)
  - libprotobuf=3.11.4=hd408876_0
  - libsodium=1.0.16=h1bed415_0
  - libstdcxx-ng=9.1.0=hdf63c60_0
  - libtiff=4.1.0=h2733197_0
  - libuv=1.40.0=h7b6447c_0
  - lightgbm=3.1.1=py37h2531618_0
  - lz4-c=1.8.1.2=h14c3975_0
  - mako=1.1.2=py_0
  - markdown=3.1.1=py37_0
  - markupsafe=1.1.1=py37h14c3975_1
  - matplotlib-base=3.1.3=py37hef1b27d_0
  - mkl=2020.0=166
  - mkl-service=2.3.0=py37he8ac12f_0
  - mkl_fft=1.0.15=py37ha843d7b_0
  - mkl_random=1.1.0=py37hd6b4f25_0
  - ncurses=6.2=he6710b0_1
  - networkx=2.4=py_1
  - ninja=1.10.2=py37hff7bd54_0
  - nltk=3.4.5=py37_0
  - numpy=1.18.1=py37h4f9e942_0
  - numpy-base=1.18.1=py37hde5b4d6_1
  - oauthlib=3.1.0=py_0
  - olefile=0.46=py37_0
  - openssl=1.1.1k=h27cfd23_0 # (updated from 1.1.1i in May 18, 2021 maintenance update)
  - packaging=20.1=py_0
  - pandas=1.0.1=py37h0573a6f_0
  - paramiko=2.7.1=py_0
  - parso=0.7.0=py_0
  - patsy=0.5.1=py37_0
  - pexpect=4.8.0=pyhd3eb1b0_3
  - pickleshare=0.7.5=pyhd3eb1b0_1003
  - pillow=7.0.0=py37hb39fc2d_0
  - pip=20.0.2=py37_3
  - plotly=4.14.1=pyhd3eb1b0_0
  - prompt_toolkit=3.0.3=py_0
  - protobuf=3.11.4=py37he6710b0_0
  - psutil=5.6.7=py37h7b6447c_0
  - psycopg2=2.8.6=py37h3c74f83_1 # (updated from 2.8.4 in May 18, 2021 maintenance update)
  - ptyprocess=0.6.0=pyhd3eb1b0_2
  - pyasn1=0.4.8=py_0
  - pyasn1-modules=0.2.8=py_0
  - pycparser=2.19=py37_0
  - pygments=2.5.2=py_0
  - pyjwt=2.0.1=py37h06a4308_0
  - pynacl=1.3.0=py37h7b6447c_0
  - pyodbc=4.0.30=py37he6710b0_0
  - pyopenssl=19.1.0=pyhd3eb1b0_1
  - pyparsing=2.4.6=py_0
  - pysocks=1.7.1=py37_1
  - python=3.7.10=hdb3f193_0 # (updated from 3.7.6 in May 18, 2021 maintenance update)
  - python-dateutil=2.8.1=py_0
  - python-editor=1.0.4=py_0
  - pytorch=1.7.1=py3.7_cpu_0
  - pytz=2019.3=py_0
  - pyzmq=18.1.1=py37he6710b0_0
  - readline=8.1=h27cfd23_0 # (updated from 7.0 in May 18, 2021 maintenance update)
  - requests=2.22.0=py37_1
  - requests-oauthlib=1.3.0=py_0
  - retrying=1.3.3=py37_2
  - rsa=4.0=py_0
  - s3transfer=0.3.4=pyhd3eb1b0_0
  - scikit-learn=0.22.1=py37hd81dba3_0
  - scipy=1.4.1=py37h0b6359f_0
  - setuptools=45.2.0=py37_0
  - simplejson=3.17.0=py37h7b6447c_0
  - six=1.14.0=py37h06a4308_0
  - smmap=3.0.4=py_0
  - sqlite=3.35.4=hdfb4753_0 # (updated from 3.31.1 in May 18, 2021 maintenance update)
  - sqlparse=0.4.1=py_0
  - statsmodels=0.11.0=py37h7b6447c_0
  - tabulate=0.8.3=py37_0
  - tk=8.6.10=hbc83047_0 # (updated from 8.6.8 in May 18, 2021 maintenance update)
  - torchvision=0.8.2=py37_cpu
  - tornado=6.0.3=py37h7b6447c_3
  - tqdm=4.42.1=py_0
  - traitlets=4.3.3=py37_0
  - typing_extensions=3.7.4.3=py_0
  - unixodbc=2.3.7=h14c3975_0
  - urllib3=1.25.8=py37_0
  - wcwidth=0.1.8=py_0
  - websocket-client=0.56.0=py37_0
  - werkzeug=1.0.0=py_0
  - wheel=0.34.2=py37_0
  - wrapt=1.11.2=py37h7b6447c_0
  - xz=5.2.5=h7b6447c_0 # (updated from 5.2.4 in May 18, 2021 maintenance update)
  - zeromq=4.3.1=he6710b0_3
  - zlib=1.2.11=h7b6447c_3
  - zstd=1.3.7=h0b5b093_0
  - pip:
    - astunparse==1.6.3
    - azure-core==1.10.0
    - azure-storage-blob==12.7.0
    - databricks-cli==0.14.1
    - diskcache==5.1.0
    - docker==4.4.1
    - gorilla==0.3.0
    - horovod==0.20.3
    - joblibspark==0.3.0
    - keras-preprocessing==1.1.2
    - koalas==1.5.0
    - mleap==0.16.1
    - mlflow==1.13.1
    - msrest==0.6.19
    - opt-einsum==3.3.0
    - petastorm==0.9.7
    - pyarrow==1.0.1
    - pyyaml==5.4
    - querystring-parser==1.2.4
    - seaborn==0.10.0
    - spark-tensorflow-distributor==0.1.0
    - tensorboard==2.3.0
    - tensorboard-plugin-wit==1.8.0
    - tensorflow-cpu==2.3.1
    - tensorflow-estimator==2.3.0
    - termcolor==1.1.0
    - xgboost==1.3.1
prefix: /databricks/conda/envs/databricks-ml
```