Deep Learning In Medicine CDSW Demo

[skiaie@cloudera.com](mailto:skiaie@cloudera.com)

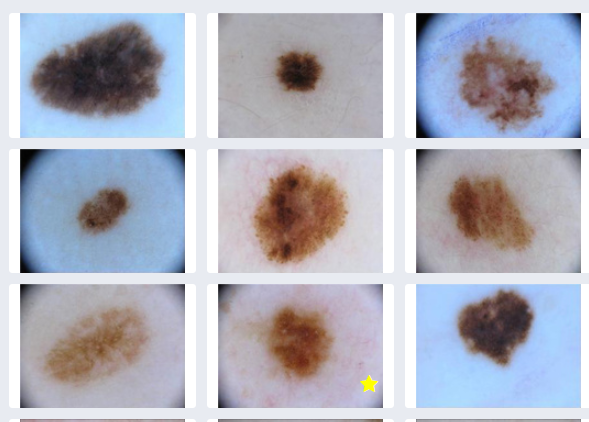
Demo Summary:

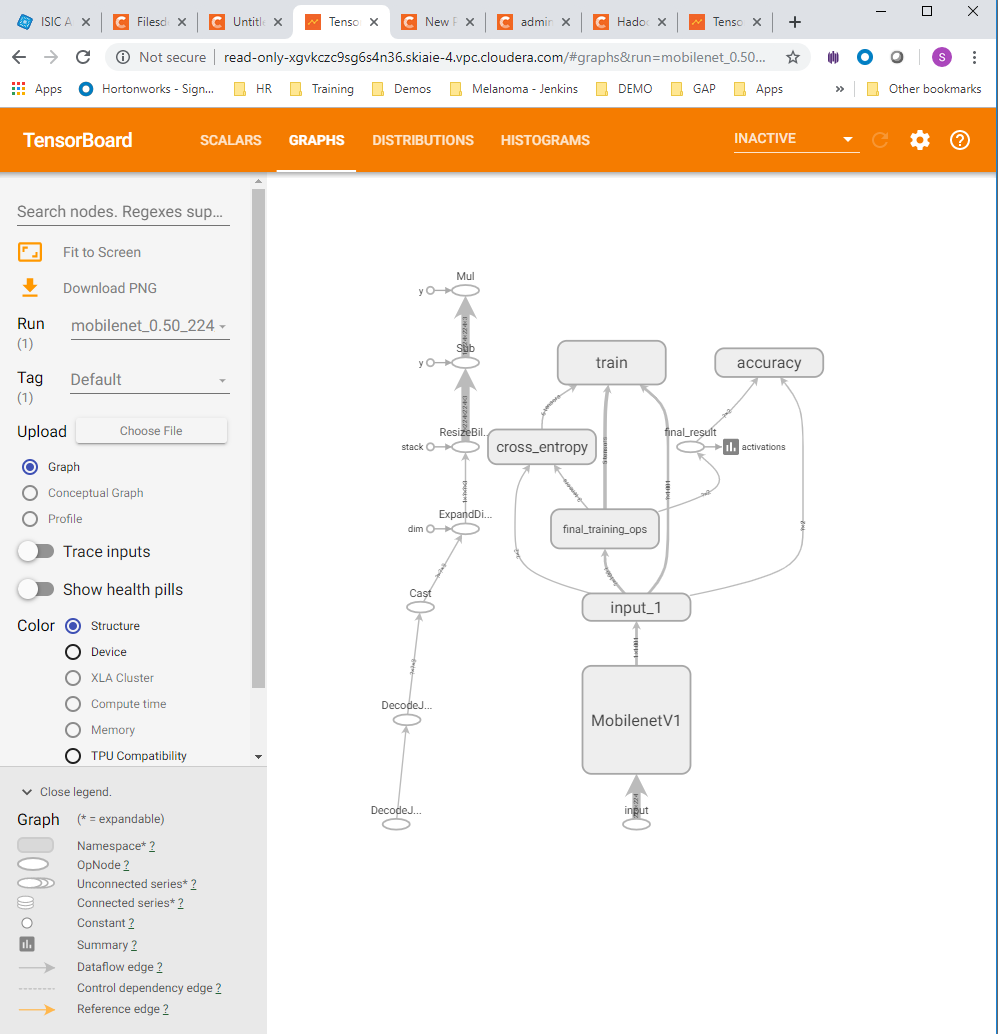
* Use Case: Diagnosing Melanoma
* Broader Healthcare Applicability:
  + Disease diagnosis using medical images
    - radiology (arteriography, mammography, radiomics)
    - dermatology
    - oncology
* Broader Industry applicability
  + Biotech
  + Pharma
  + Semiconductor Fabrication

Story

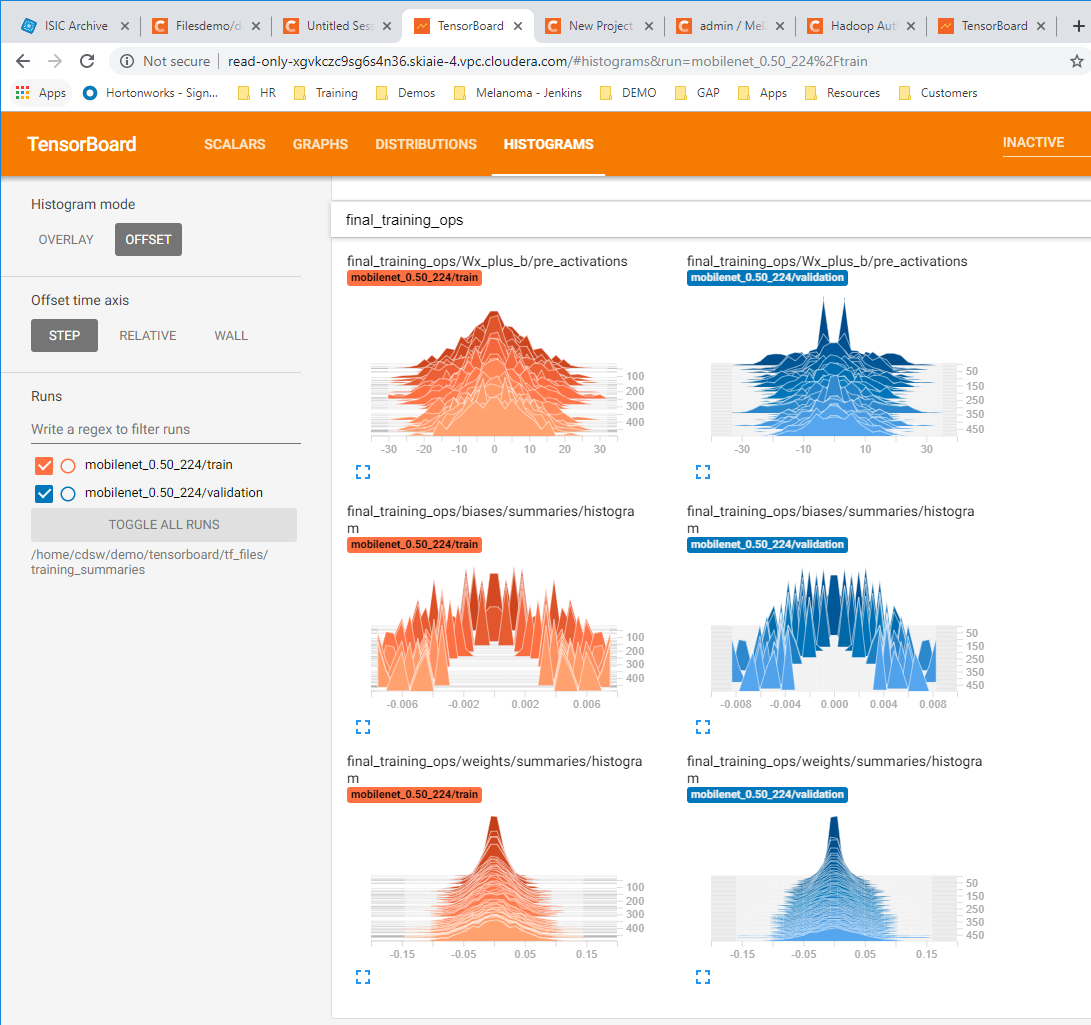
1. We take open source images of skin lesions, and use those to build a classifier to detect malignant skin lesions
2. We evaluate the performance of our model using TensorBoard, and matplotlib in CDSW
3. We deploy the model onto a mobile device for use in clinical settings
4. We use our mobile app to determine if a patient needs critical attention from a physician (Note: in the demo we use a model deployed on a mobile device, for simplicity. I.e. inference happens on the edge, using a low latency, MobileNet model. The more likely choice for this use case would be to perform classification in batch or perform the inference centrally, using a model with superior performance characteristics (measured by AUC).

**1.**

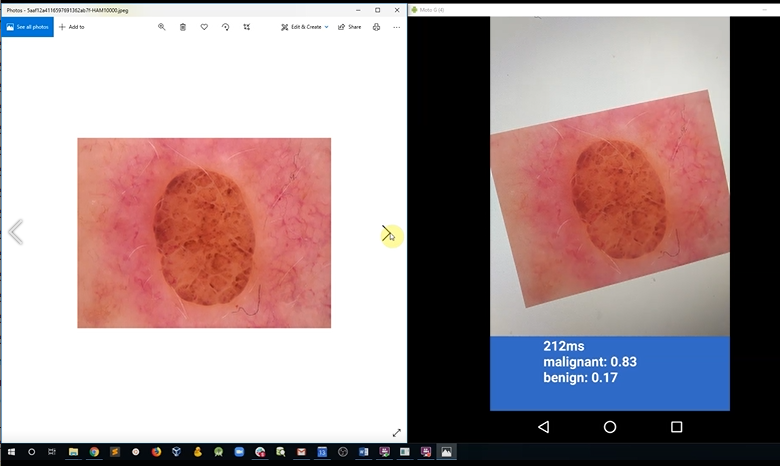




**2.**



**3,4.**



The preliminary talk track videos are here:

* [Deck Talk Track](http://bit.do/fjHDB)
* [Demo Talk Track](http://bit.do/fjHJE)

The talk tracks here were made for brevity, and to focus on the specifics of this use case, so some of the boilerplate CML messaging is missing (e.g. some of the collaborative features).  I plan to add that.

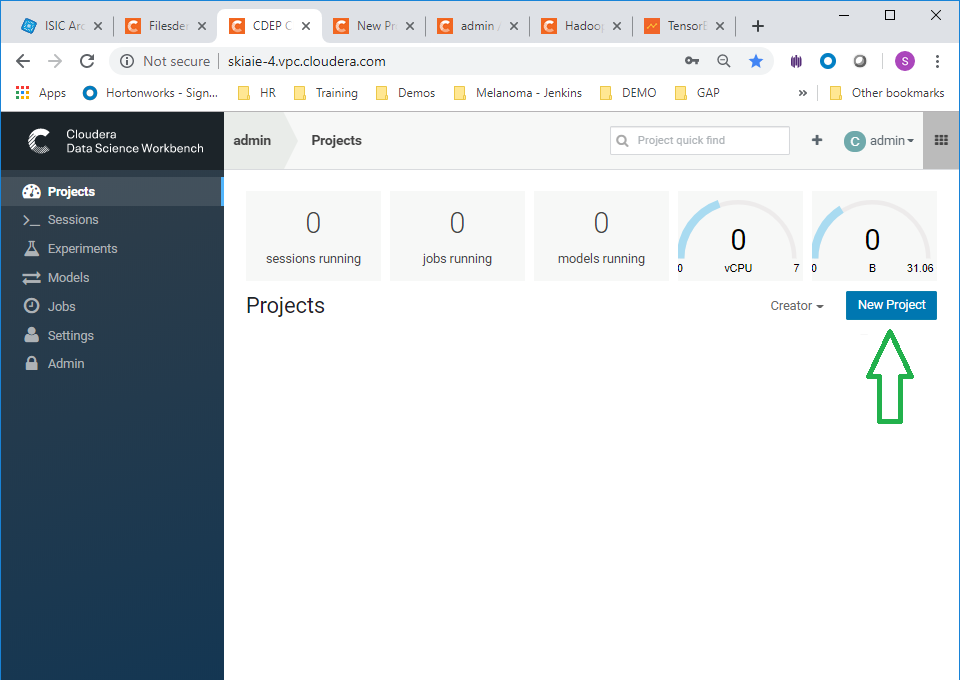
The deck is here:

* [Deck](http://bit.do/fjHJf)

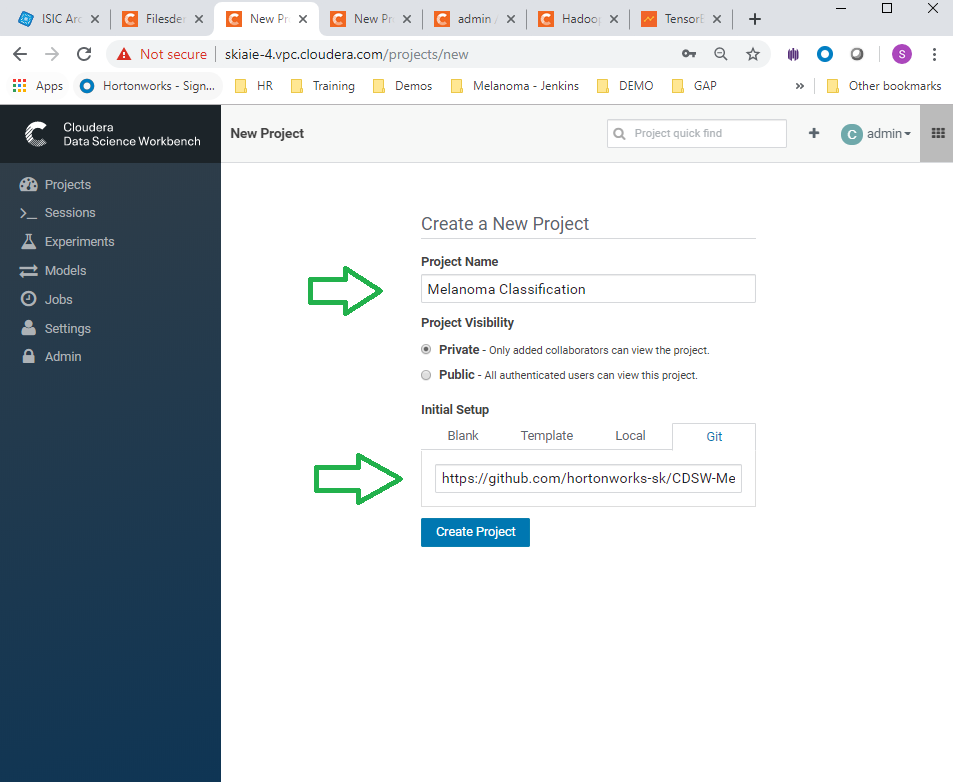
Demo Setup

The setup takes 5 minutes

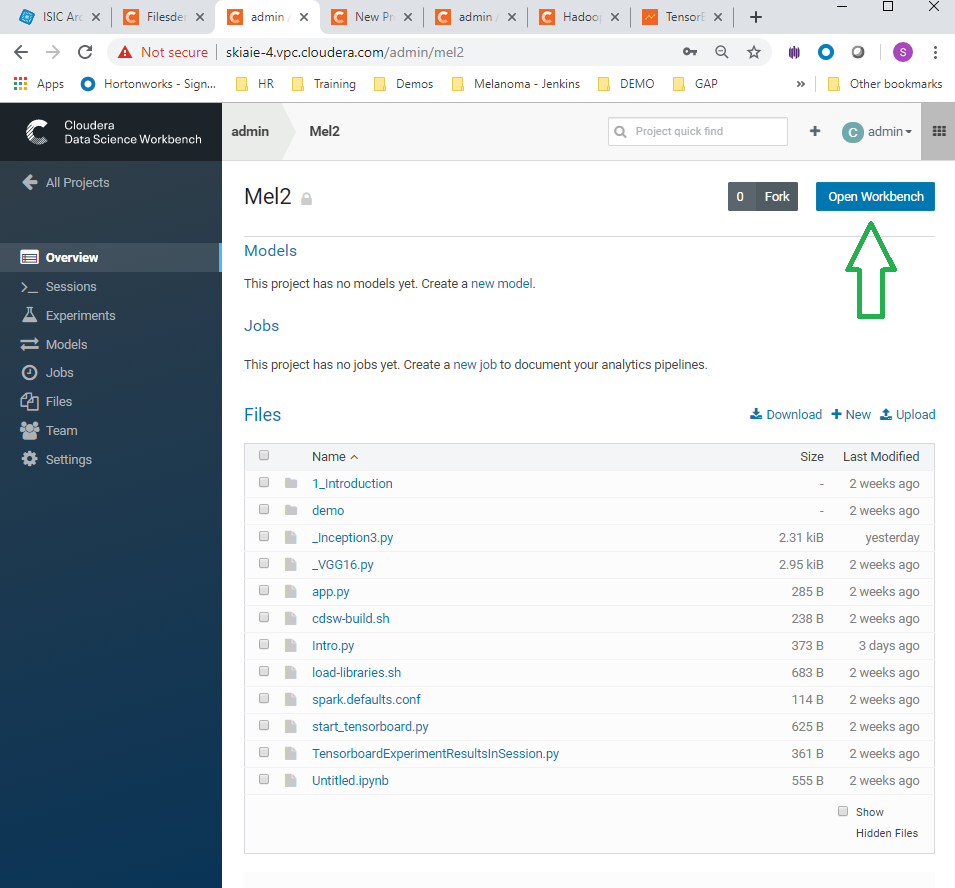
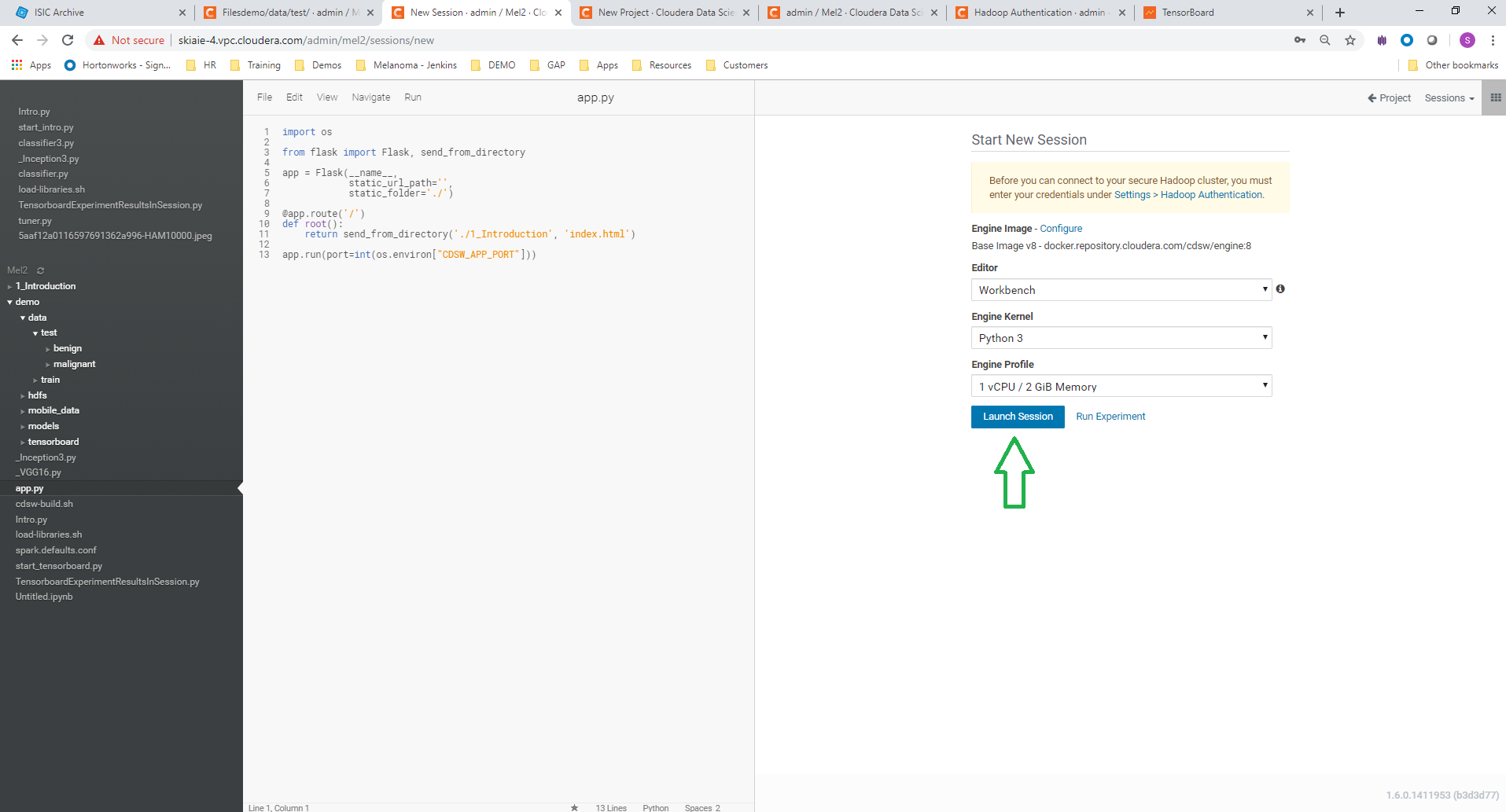
1. In CDSW Go to Projects, and create a New Project



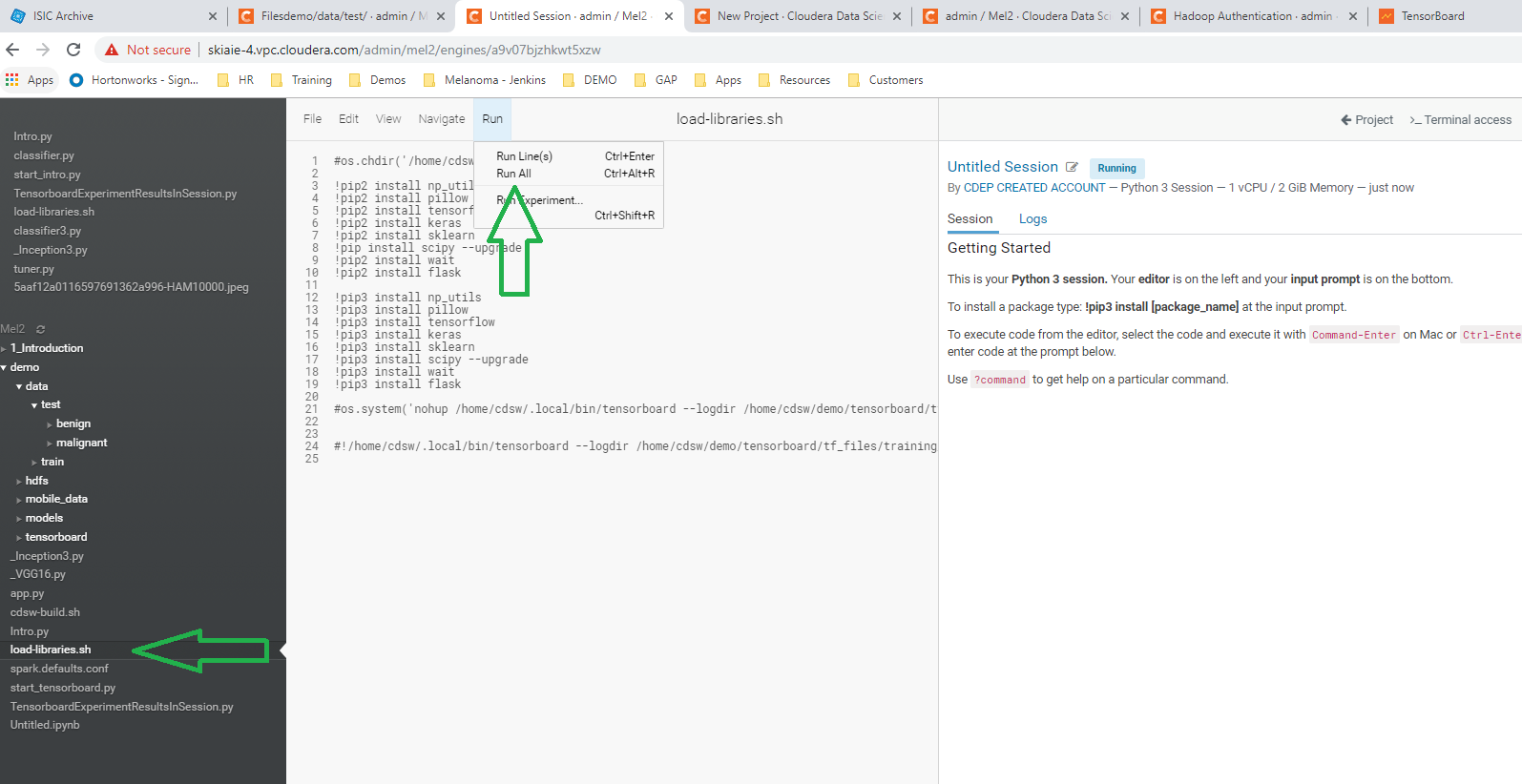
1. Name the Project “Melanoma Classification”, and in the initial setup use git repo: <https://github.com/hortonworks-sk/CDSW-Melanoma2.git> , and hit the create button



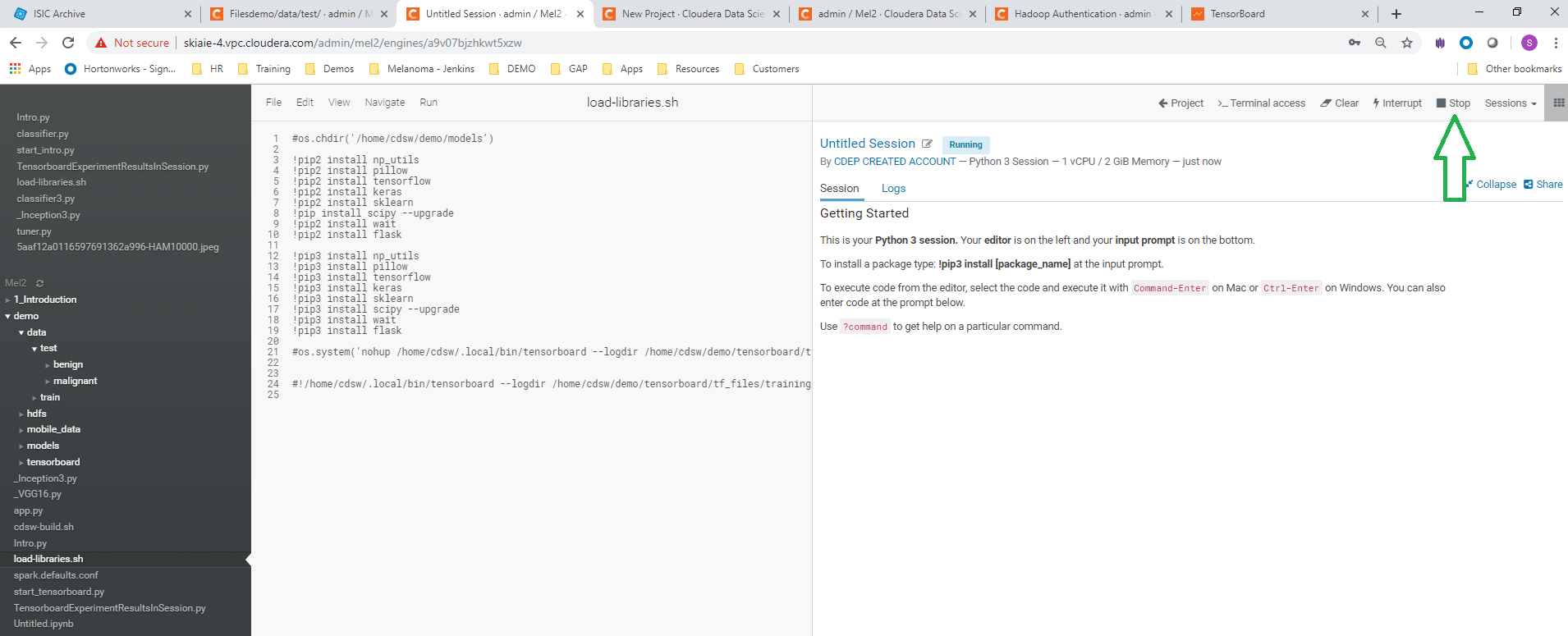
1. Launch a Python 3 workbench session

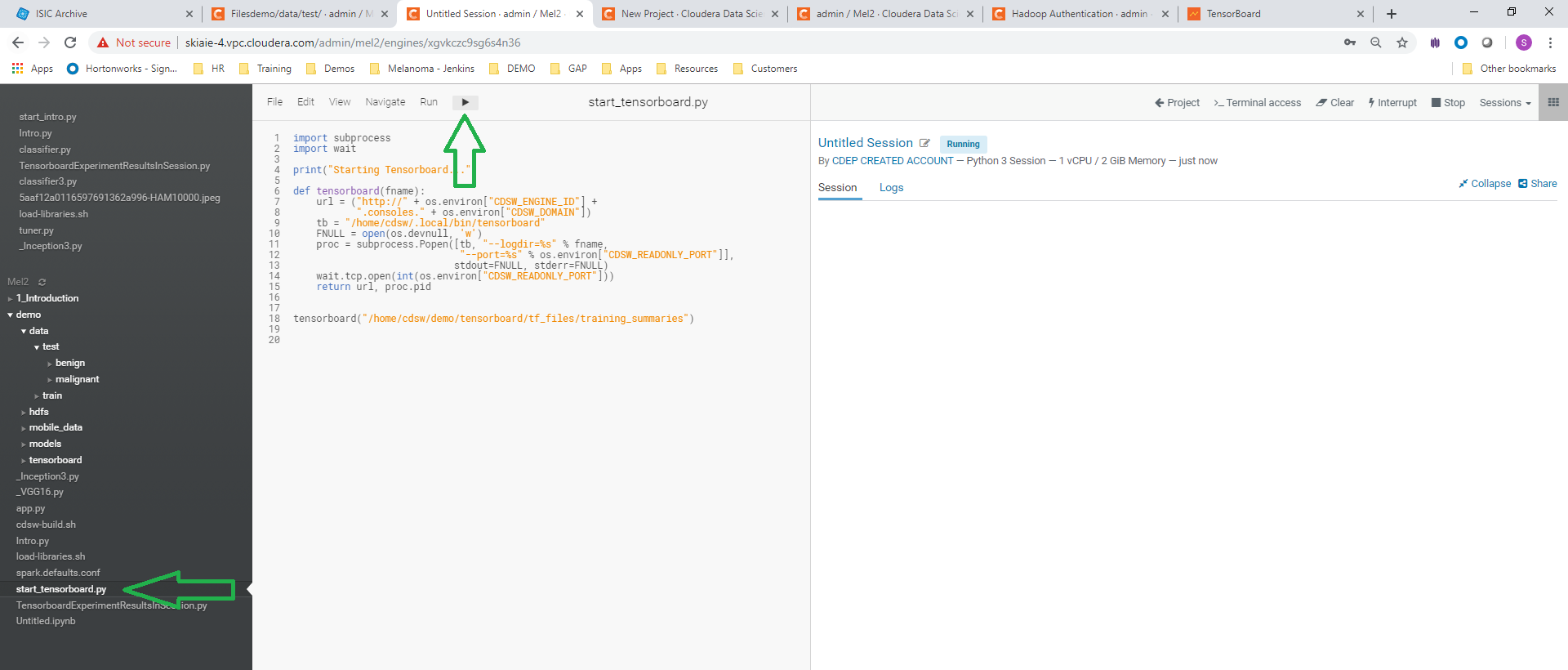
1. Navigate to the **load-libraries.sh** script, and run the script. This will load the libraries needed for the demo.



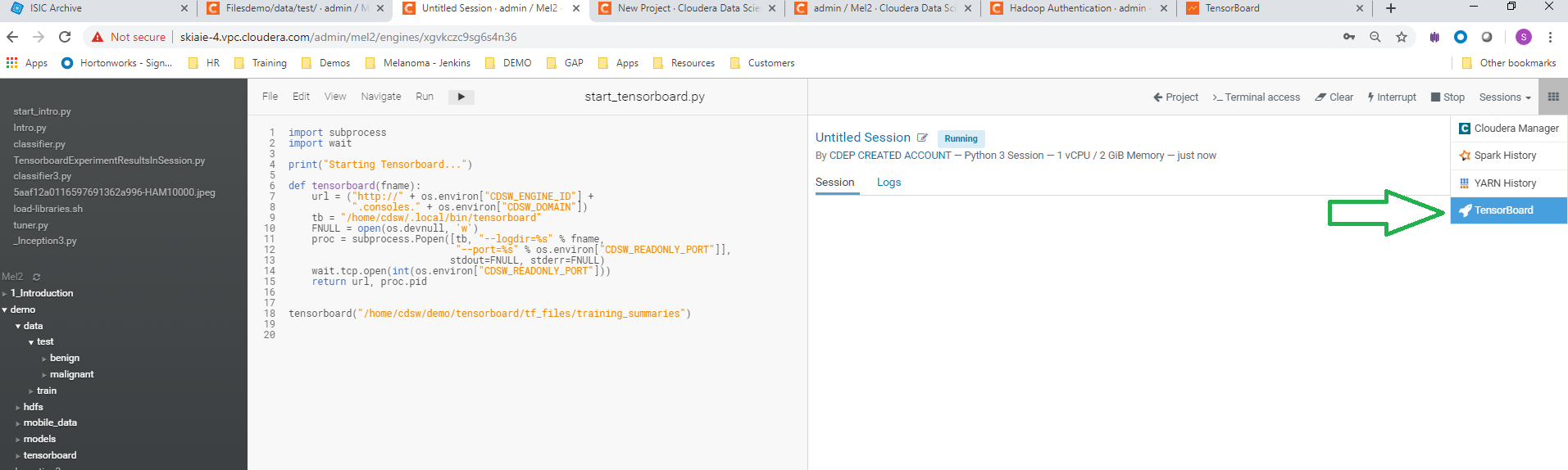
1. Stop the Python 3 workbench session, and open another Python 3 session. This is required for some of the libraries to be available.



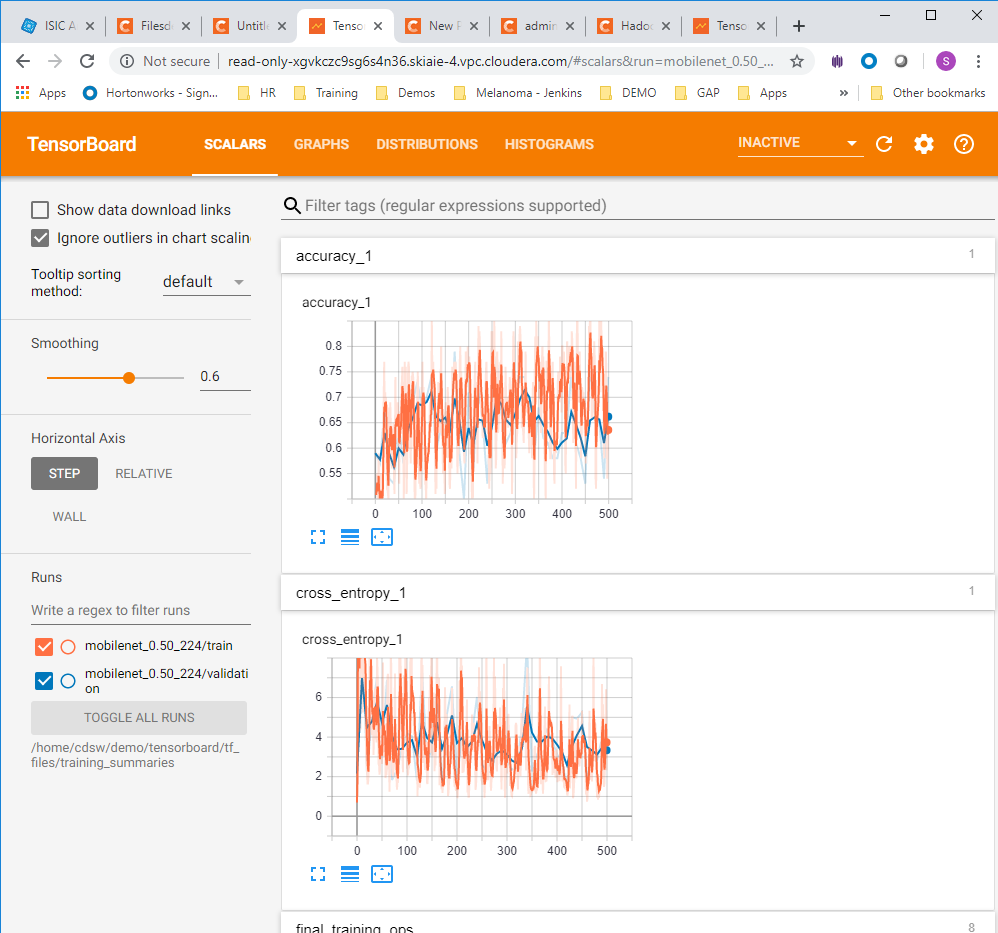
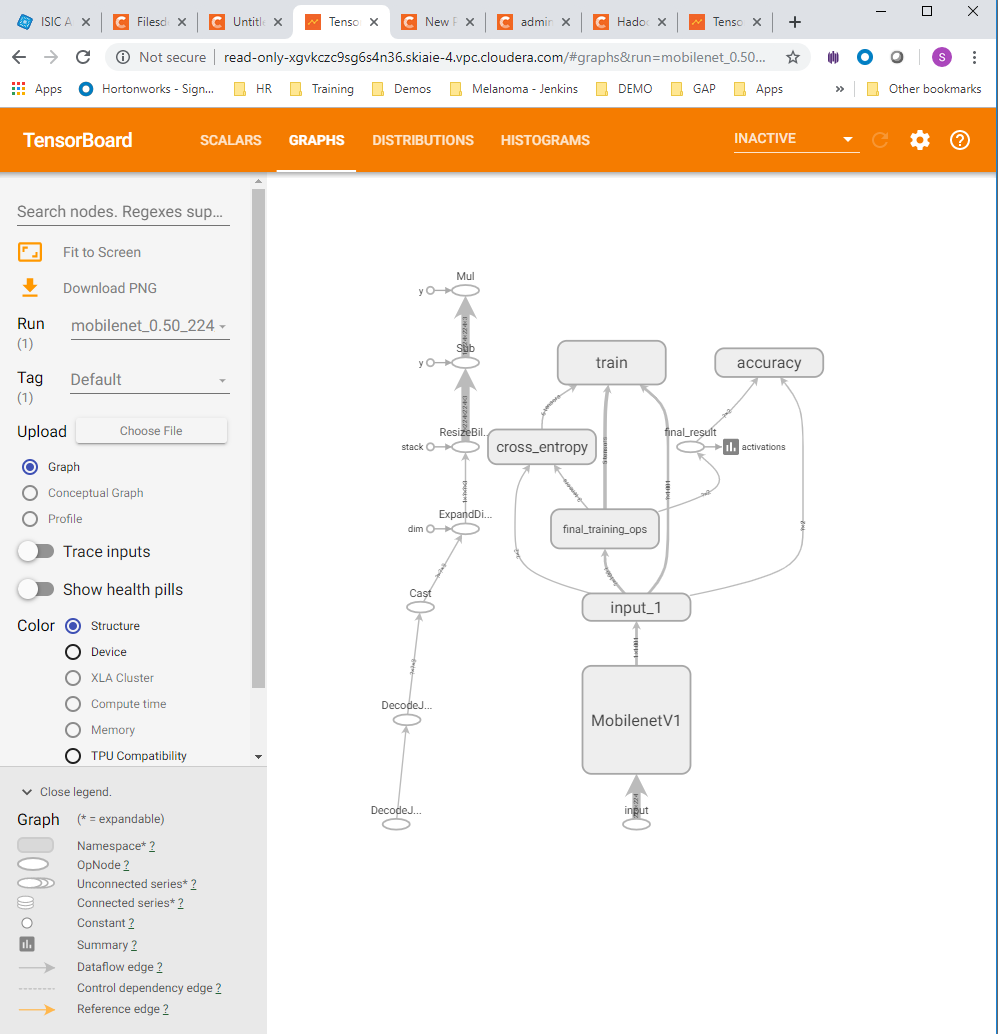
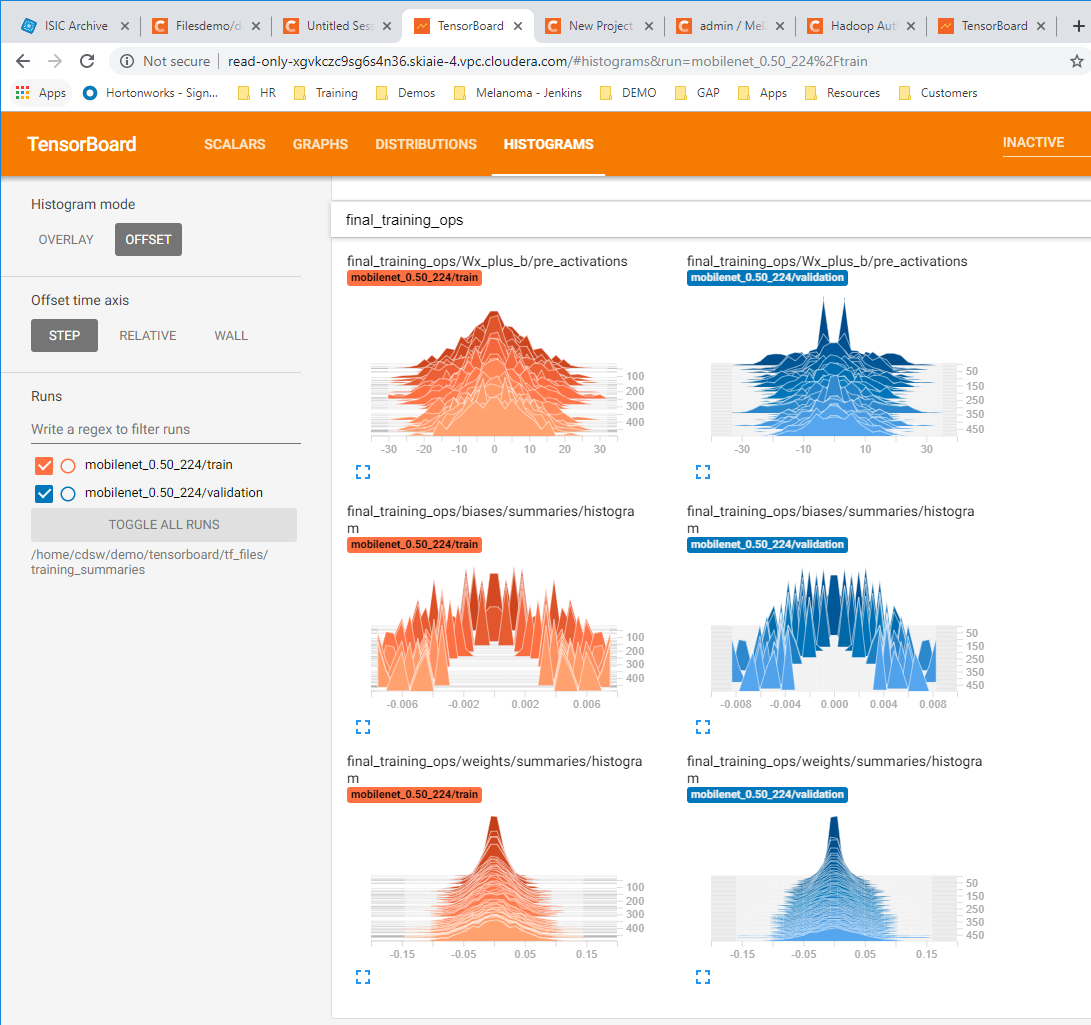
1. Navigate to the **start\_tensorboard.py** script, and run this.



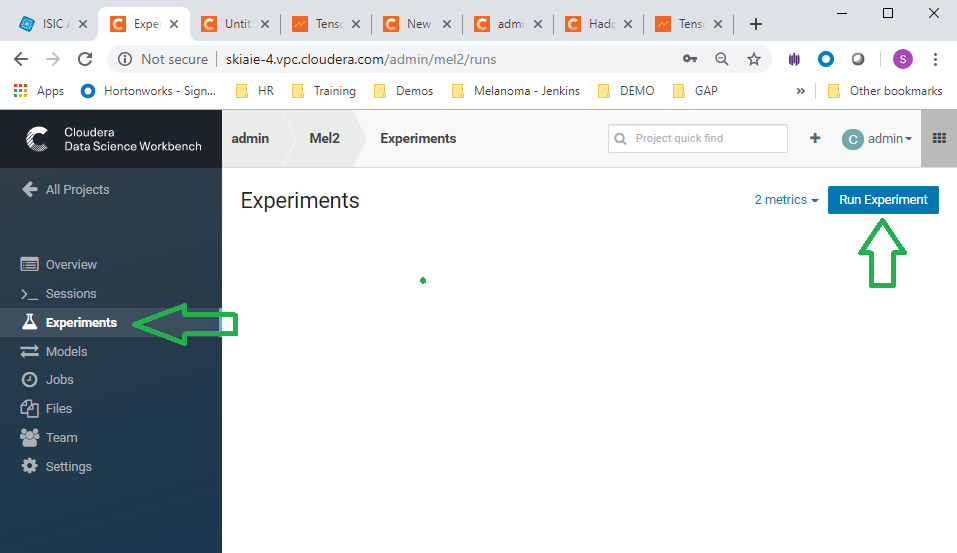
1. Check that the Tensorboard link is displaying in CDSW and that tensorboard is running, by clicking the tensorboard link



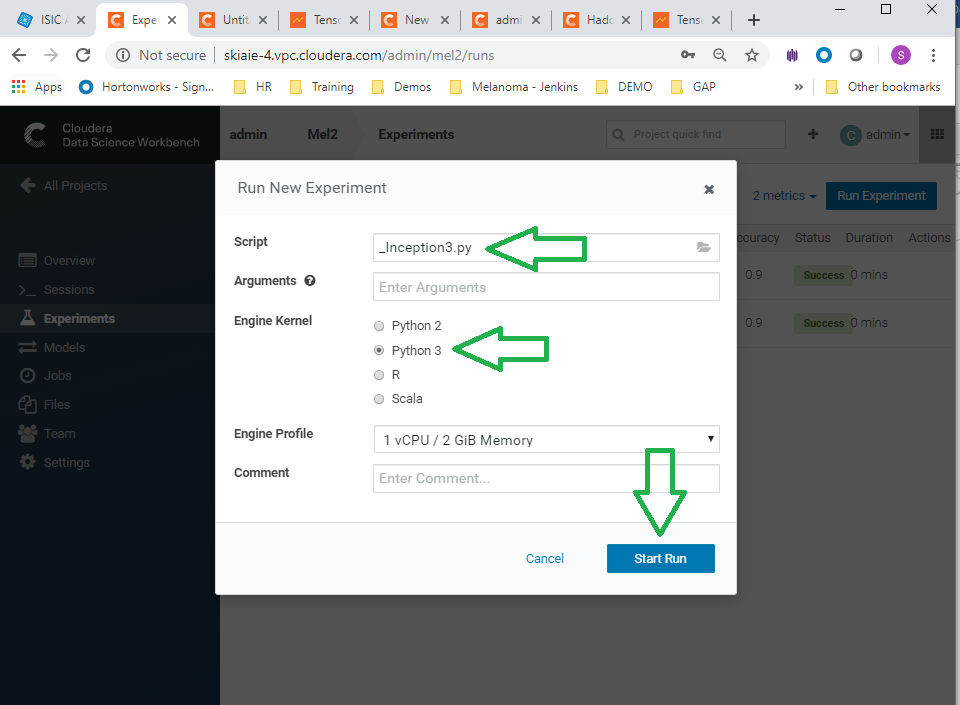
1. Click on the tensorboard tabs for **Scalars**, **Graph** and the **Histograms**, to check that these are displaying correctly (each are shown in order below)

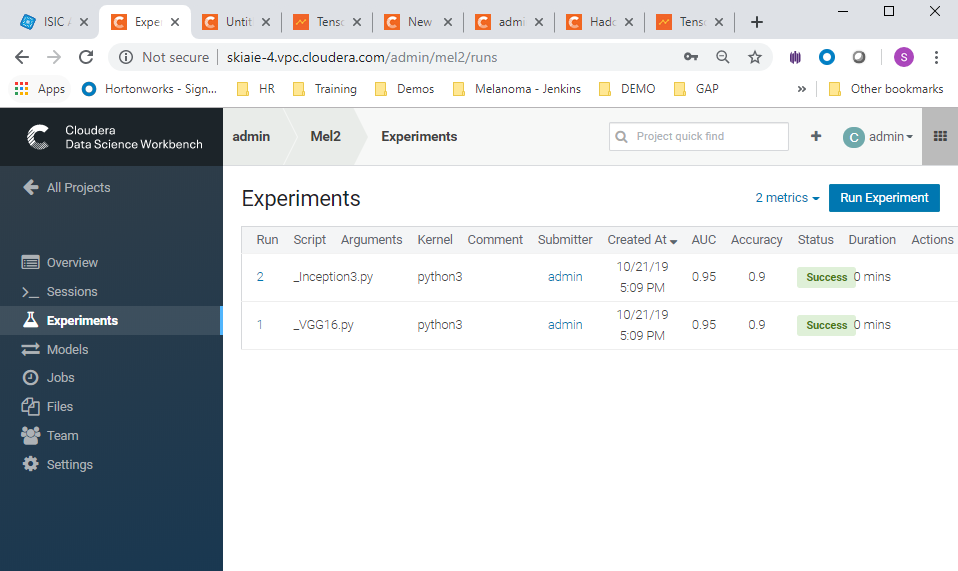
1. Navigate to experiments and click run experiment



1. Run experiments for the **\_Inception3.py**, and **\_VGG16.py, scripts.** Use the python 3 kernel. No need to supply arguments for these.



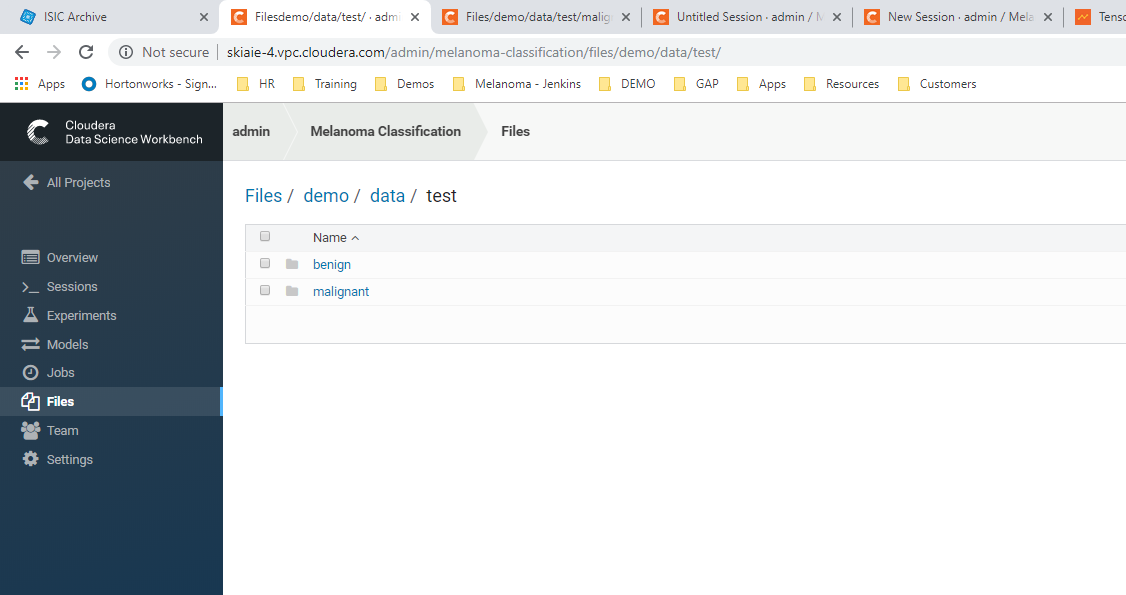
1. When these runs have completed, you should see the experiments listed as successful in the experiments view (as in the screenshot below)



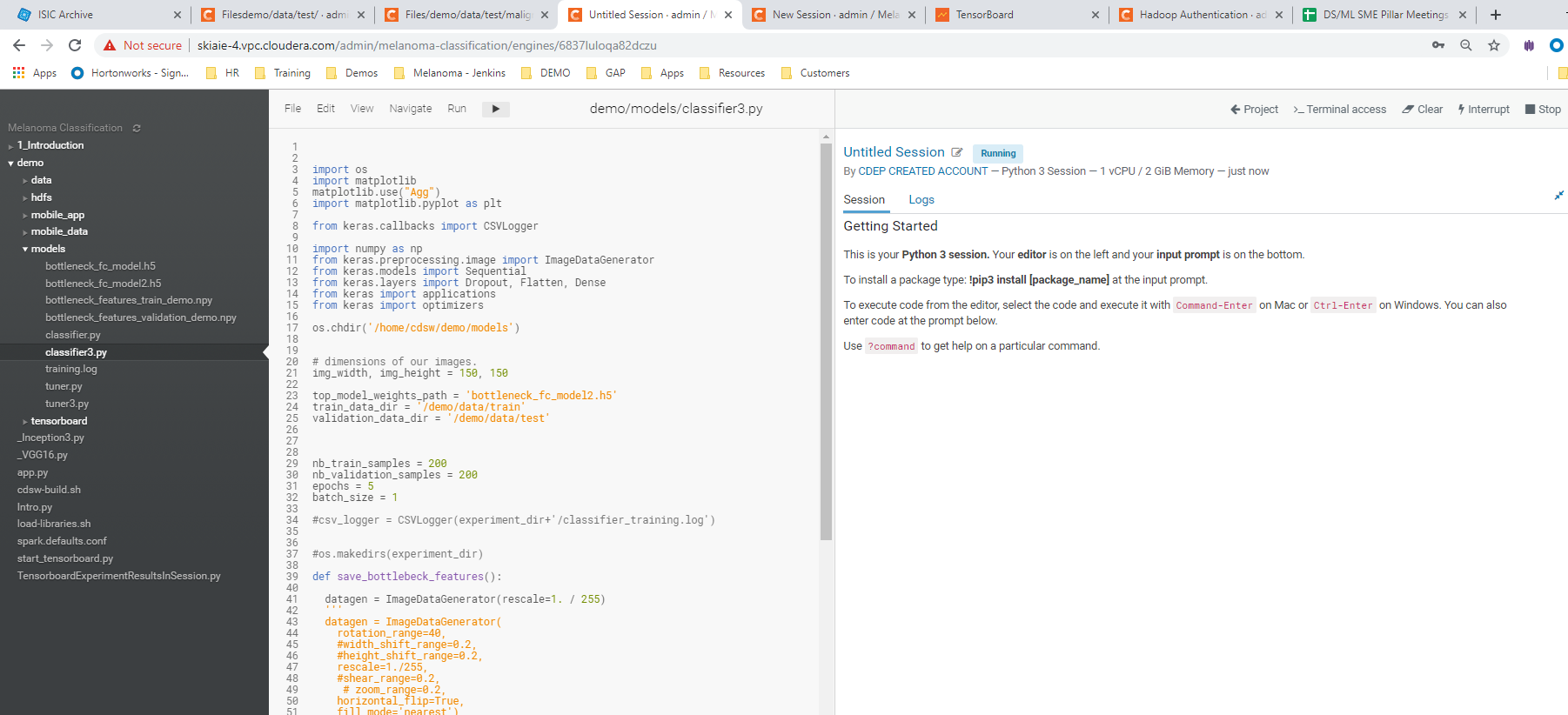
Pre-Demo Setup:

Having the following tabs open, in a Chrome window, may be useful (these are the tabs open in the talk track video):

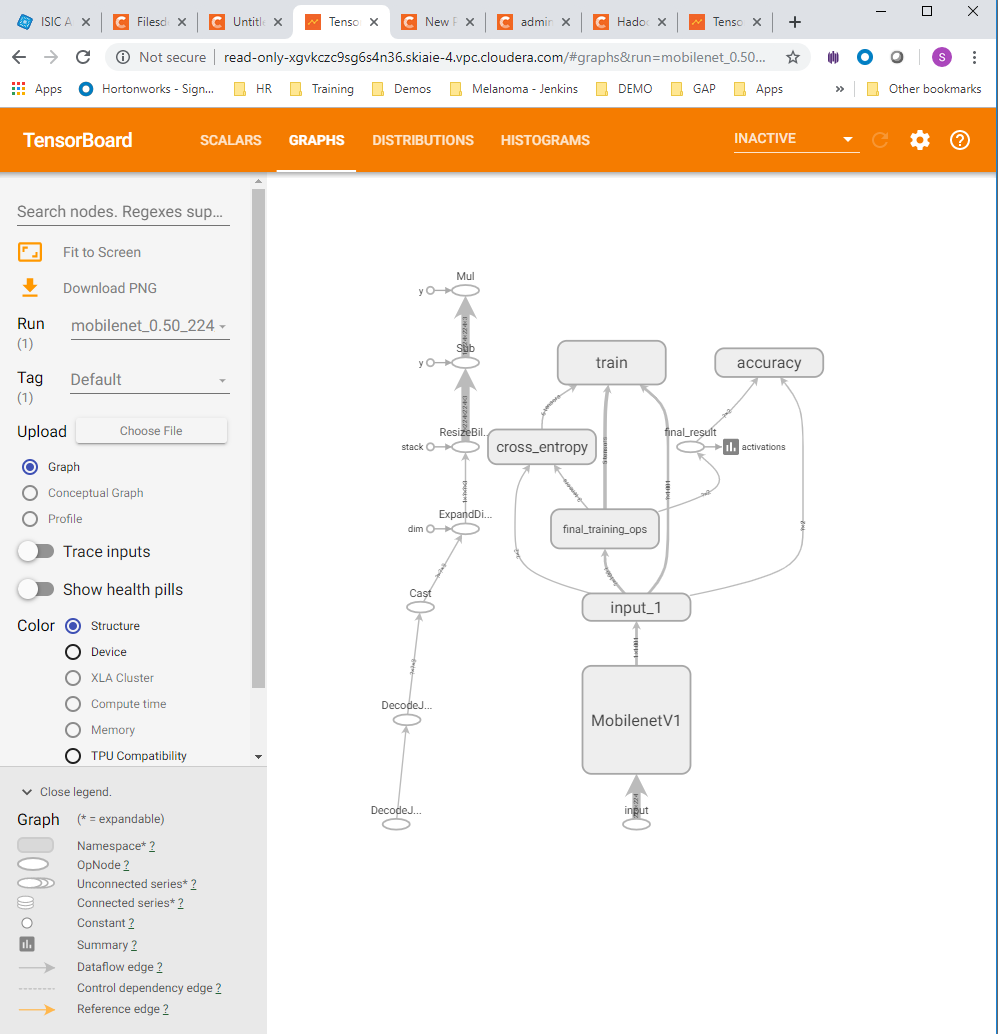
1. The ISIC dataset homepage (<https://www.isic-archive.com/#!/topWithHeader/wideContentTop/main>)
2. The CDSW file view of the training data folder <http://your-cdsw-host.and-domain.com/yourusername/melanoma-classification/files/demo/data/test/> (This is at the folder path: **demo > data > test** in CDSW)



1. A Python 3 workbench session (loaded within the Classifying Melanoma project) pointing to the script to train the classifier (This is at the path: **demo > models > classifier3.py**, in CDSW)



1. Tensorboard, with the Graph view



1. The new Projects <http://your-cdsw-host.and-domain.com/projects/new>
2. The experiments page
3. [http://[your-cdsw-host.and-domain.com]/projects/new/[your-username]/mel2/runs](%20http://%5byour-cdsw-host.and-domain.com%5d/projects/new/%5byour-username%5d/mel2/runs) admin/mel2/runs