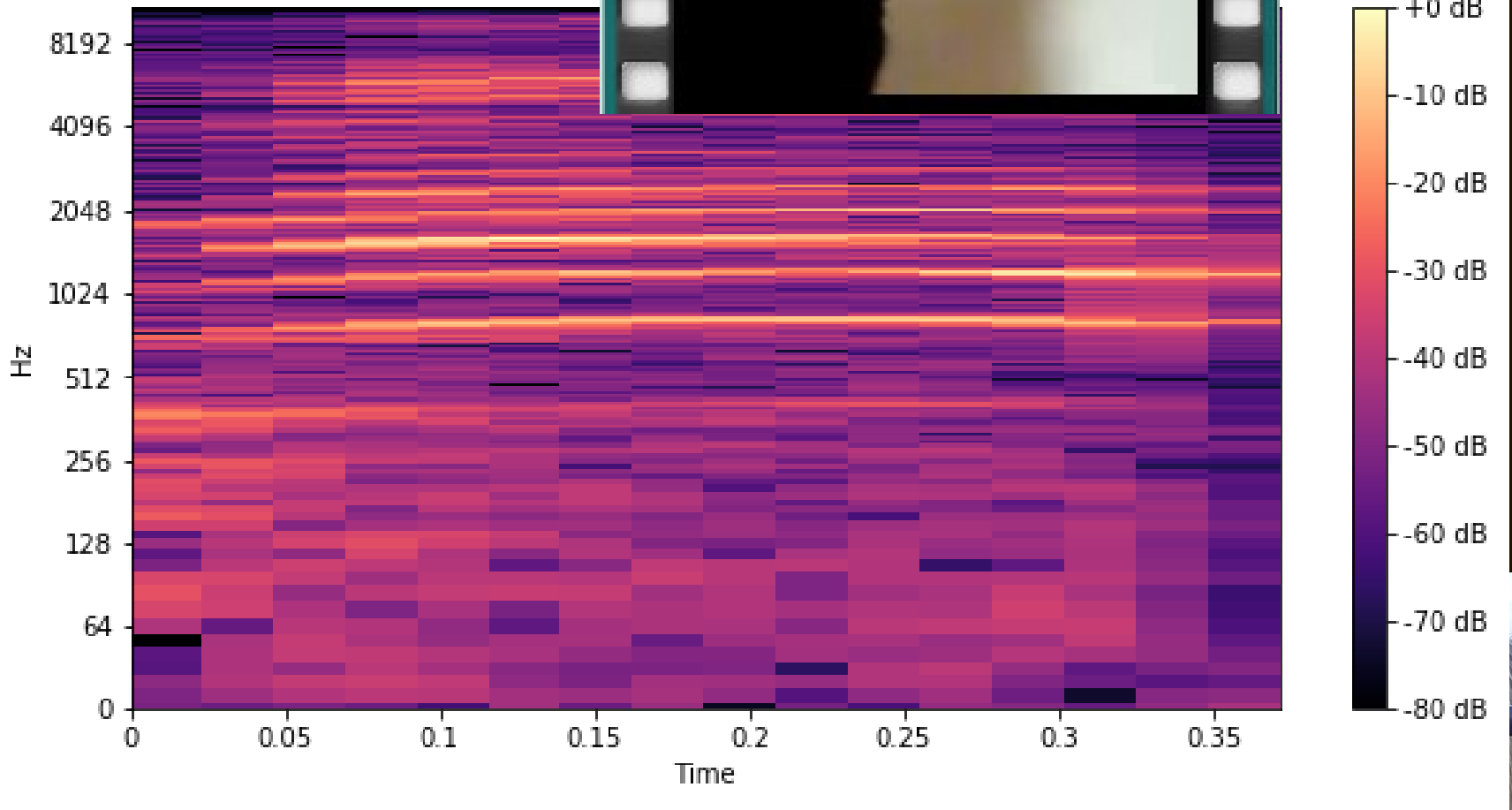


Building AI Models with IBM Watson Studio

Sarah Packowski & Wendy Switzer
Monday, October 29, 2018



Agenda

1. Introduction to Watson Studio
2. Importance of *open data* for AI
3. Description of the 4 workshop samples
4. All together: work through the Flags sample
5. At your own pace: work through one or more of the other samples

Workshop goals

- **Have fun**
 - Interesting & entertaining samples
- **Learn**
 - IBM Watson Studio
 - Visual Recognition
 - Python notebooks
 - Python app (local, IBM Cloud)
- This is not intended to be:
 - AI best practices
 - Deep dive into building neural networks by hand

For a deep dive, full-day workshop on Wednesday:

Practical machine learning with Python

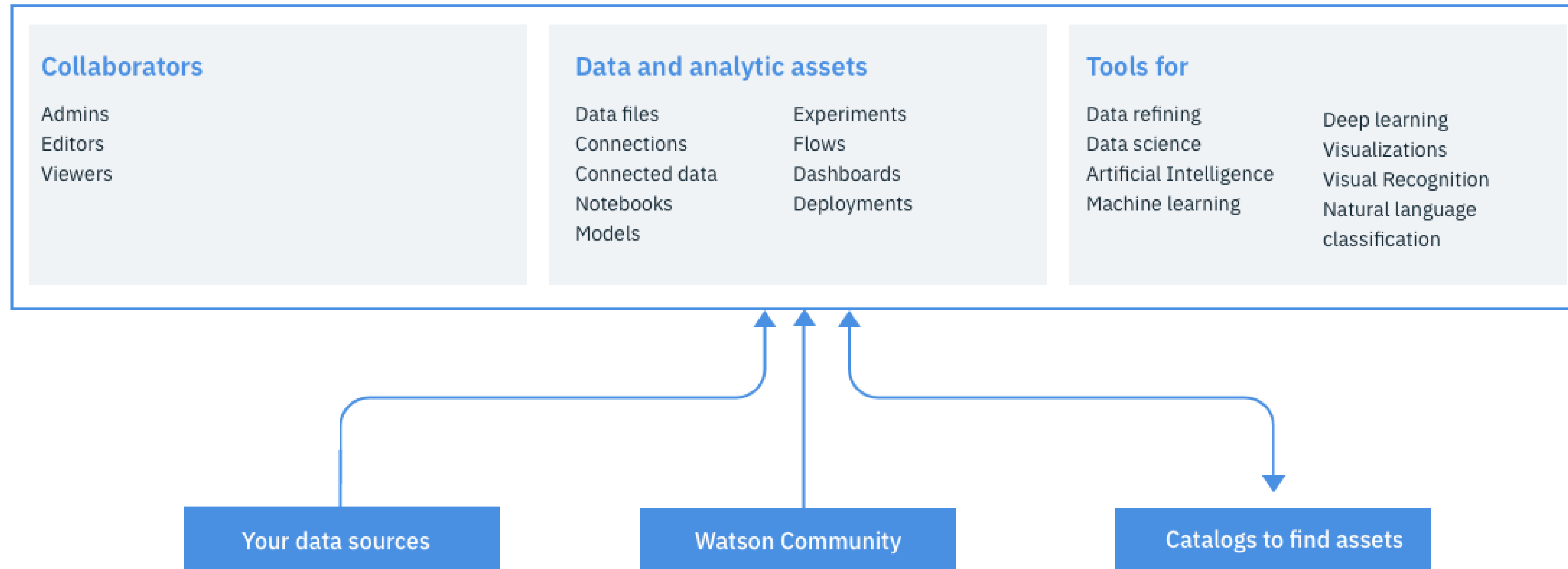
<https://www-01.ibm.com/ibm/cas/cascon/workshopsignup/displayWorkshop?Num=500>

Introduction to IBM Watson Studio

- Provides you with the environment and tools to solve your business problems by collaboratively working with data.
- You can choose the tools you need to:
 - analyze and visualize data
 - cleanse and shape data
 - ingest streaming data
 - create, train, and deploy machine learning models

Introduction to IBM Watson Studio

Projects: work with data



Let's have a look at Watson Studio: <https://dataplatform.cloud.ibm.com>

Watson Studio videos: <https://dataplatform.cloud.ibm.com/docs/content/getting-started/videos.html>

Importance of *open data*

- The idea behind open data is that some data should be:
 - free
 - available to everyone
 - available to be used, republished, with no restrictions
- Need lots of data for AI
- When searching, might not find exactly what you are looking for
- If you do find data, you need to ensure you understand the terms of use in relation to your usage of that data
- Waston Studio community is a great starting point for open

4 workshop samples

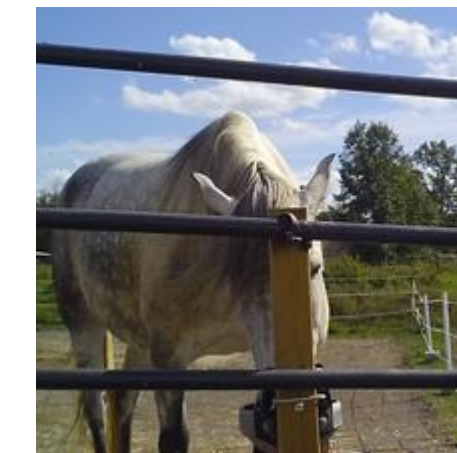
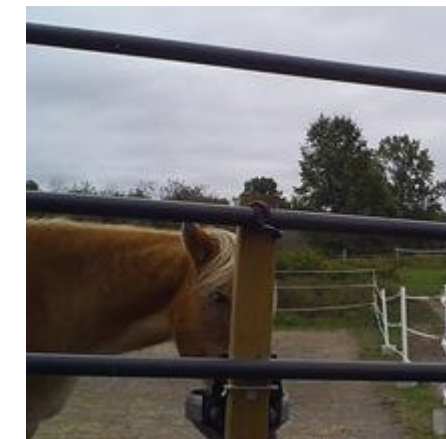
Flags

[\[link\]](#)



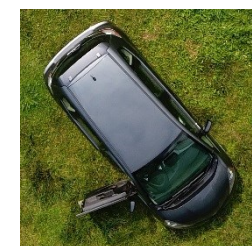
Water Consumption

[\[link\]](#)



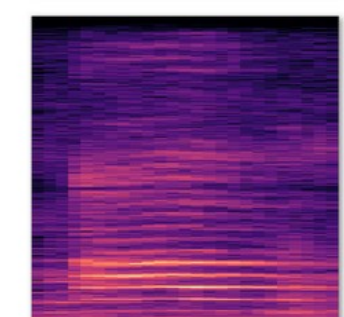
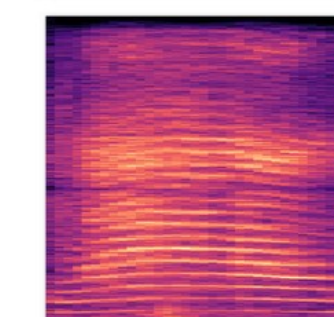
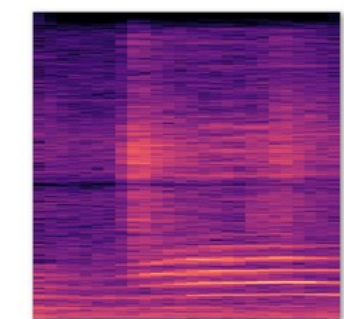
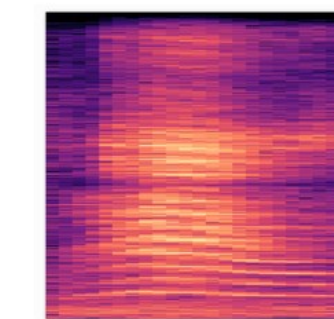
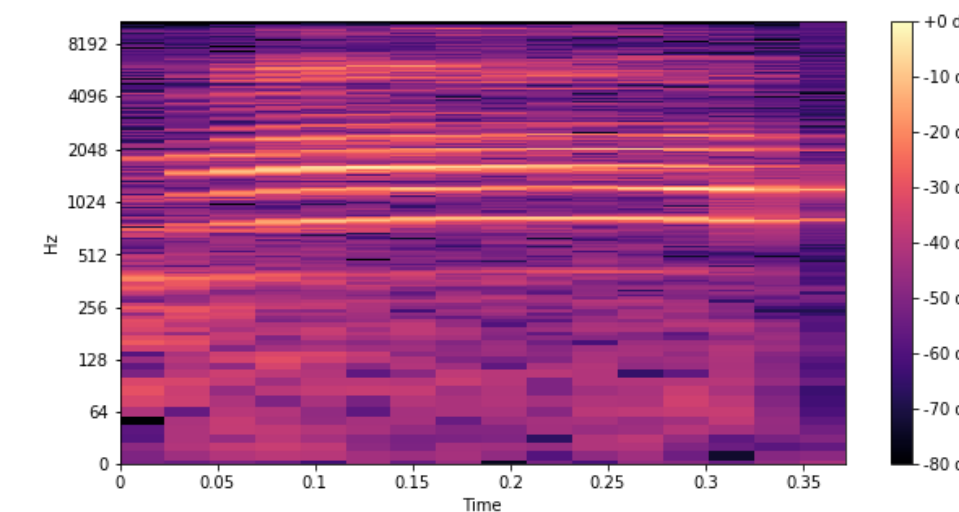
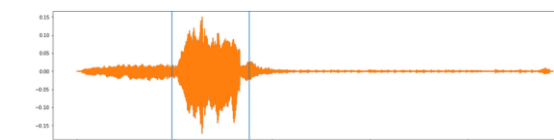
Search & Rescue

[\[link\]](#)



Cat Translator

[\[link\]](#)



All 4 samples follow the same pattern

Same 6 sections:

1. Train model
2. Test model (GUI)
3. Test model (Python notebook)
4. Explore “app code” in a notebook
5. Run local Python app
6. Push app to IBM Cloud

README

How to train the Flags model

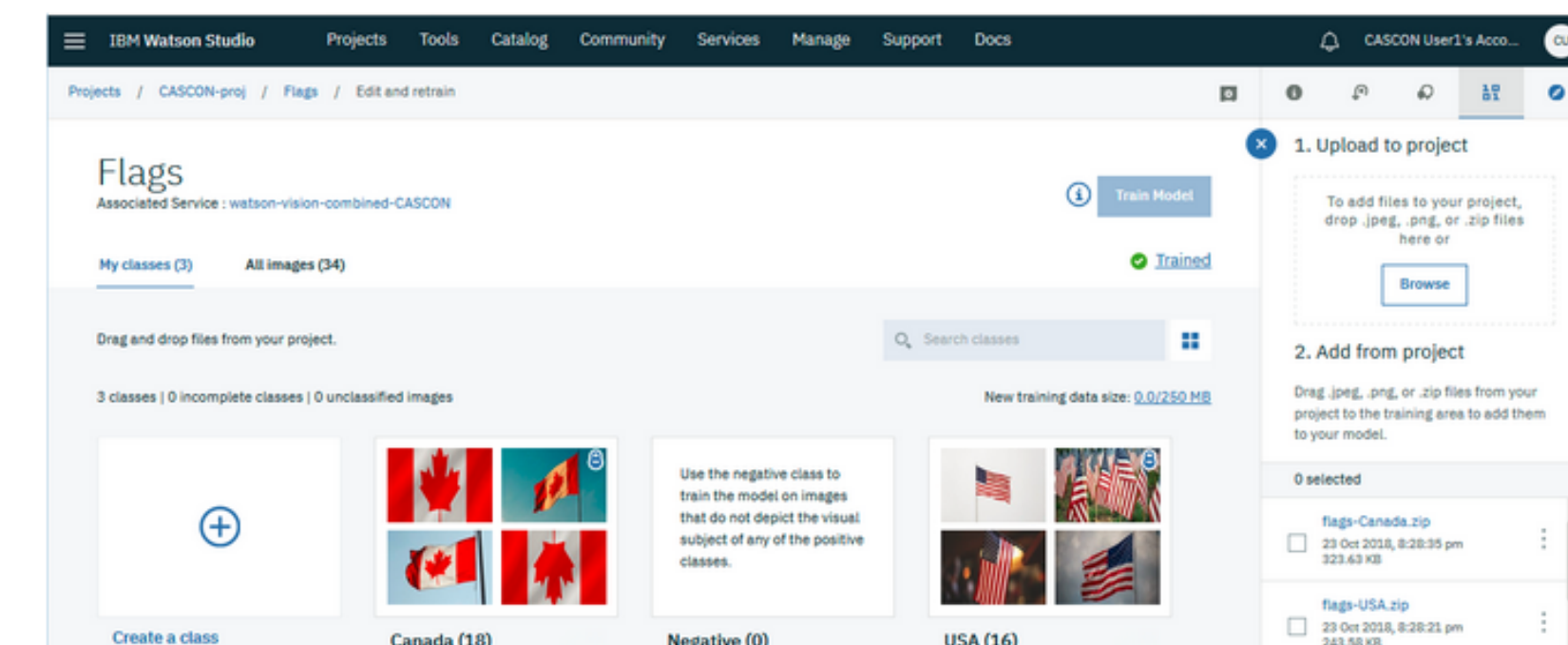
Training files

- flags-Canada.zip
- flags-USA.zip

Class names

The sample notebooks and apps expect the classes to be named the same as the training files, but without the 'flags-' prefix and without the .zip extension:

- Canada
- USA



Prerequisites

Required

1. Download GitHub repo:
https://github.com/spackows/CASCON-2018_Analyzing_images/
2. Set up Watson Studio on IBM Cloud:
<https://dataplatform.cloud.ibm.com/registration/stepone>
3. Create a project in Watson Studio
4. Add training data as **Assets** to your project:
cat-translator/training-data/*.zip
flags/training-data/*.zip
search-and-rescue/training-data/*.zip
water-consumption/training-data/*.zip

Video demonstrating these steps:
<https://youtu.be/rBD5MjqznNY>

Optional

1. To run sample apps locally, install Python:
<https://www.python.org/downloads/>
 - Make sure the installer adds environment variables
 - Mac users, also install pip:

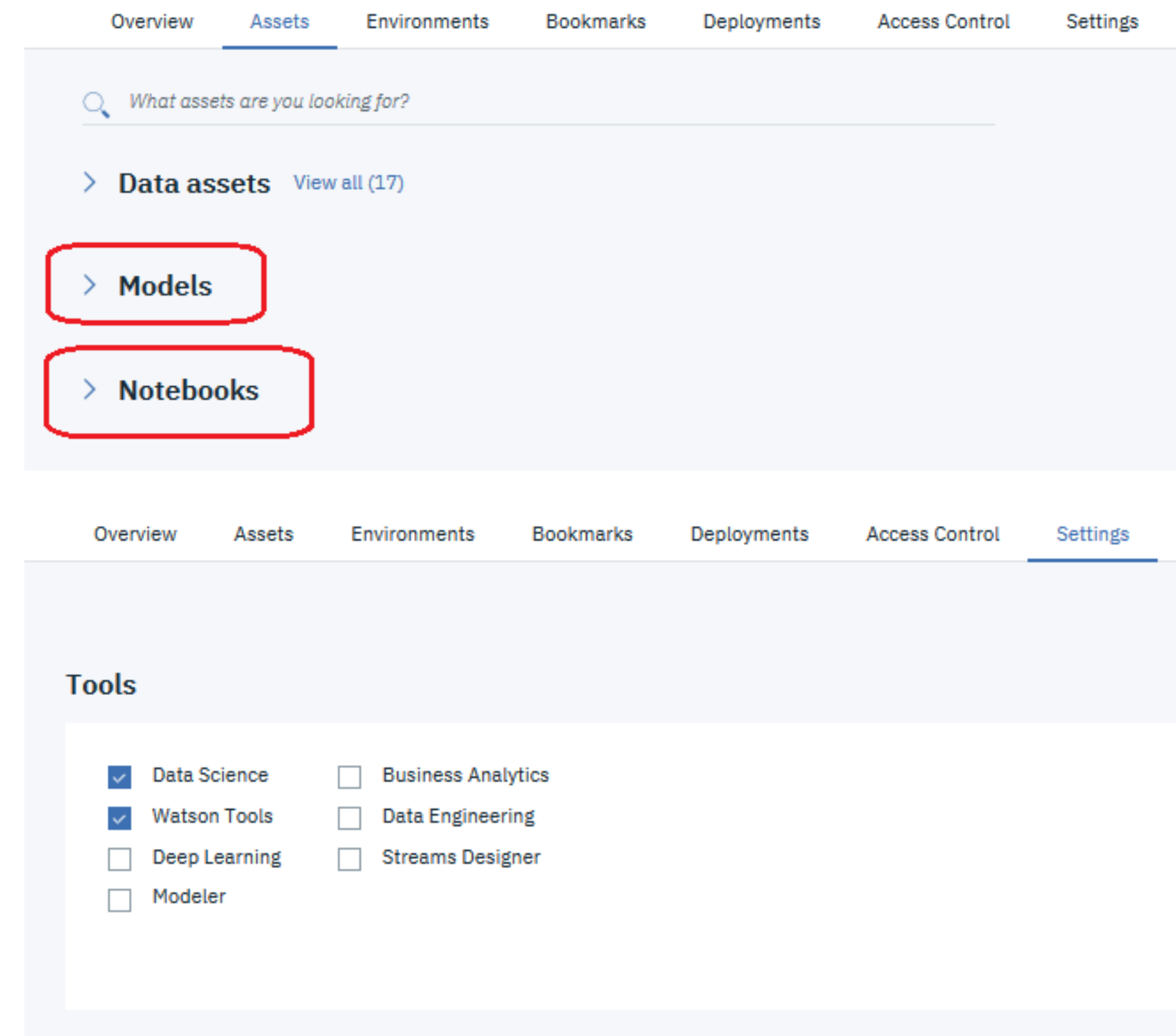
```
> sudo easy_install pip
```
 - Mac users, also add your user base to your path
 - Look up the user base directory:

```
> python -m site --user-base
```
 - Add that directory to the file `/etc/paths`
2. To push the apps to IBM Cloud, install the CLI:
https://console.bluemix.net/docs/cli/reference/ibmcloud/download_cli.html#install_use

Add tools to the project

If you cannot see a Models section and a Notebooks section in the **Assets** tab of your project...

Add “Data Science” and “Watson Tools” in the **Settings** tab.



Video demonstrating these steps: <https://youtu.be/J4YibvrevfQ>

Flags sample

https://github.com/spackows/CASCON-2018_Analyzing_images/tree/master/flags

1. Train model (GUI tool in Watson Studio)
2. Test model (GUI tool in Watson Studio)
3. Test model (Python notebook in Watson Studio)
4. Explore Python “app code” in a notebook
5. [Optional] Run local Python app
6. [Optional] Push app to IBM Cloud

[Flags - Section 1] Train model

1. On the **Assets** tab of your project in Watson Studio, click “New Visual Recognition model”
2. When prompted, associate an instance of the Visual Recognition service with your project (this doesn’t happen if you have an instance already)
3. When the model builder tool loads, give your new model a name
4. From the data panel, add `flags-Canada.zip` and `flags-USA.zip` to the model
5. Adjust the names of the classes to “Canada” and “USA”
6. Click **Train Model**

Video demonstrating these steps: <https://youtu.be/KtHe4NLaEeE>

[Flags - Section 2] Test model (GUI)

1. Navigate to the test area of the Visual Recognition model builder in Watson Studio
2. Test some sample images from the GitHub repo you downloaded in the subdirectory:
`flags/test-data`

Video demonstrating these steps: <https://youtu.be/k6dmyiOTMYw>

[Flags - Section 3] Test model (notebook)

1. On the **Assets** tab of your project, click “New notebook”
2. Click “From URL”
3. Enter the URL of the test notebook in GitHub:
https://github.com/spackows/CASCON-2018_Analyzing_images/blob/master/flags/sample-notebooks/flags-test-notebook.ipynb
4. Give the notebook a name (Example: flags-test-notebook)
5. Click **Create Notebook**
6. Run the cells in the notebook

Video demonstrating these steps: <https://youtu.be/J7lgL5CwKl8>

[Flags - Section 4] “App code” in a notebook

1. On the **Assets** tab of your project, click “New notebook”
2. Click “From URL”
3. Enter the URL of the test notebook in GitHub:
https://github.com/spackows/CASCON-2018_Analyzing_images/blob/master/flags/sample-notebooks/flags-app-code-notebook.ipynb
4. Give the notebook a name (Example: flags-app-notebook)
5. Click **Create Notebook**
6. Run the cells in the notebook

Video demonstrating these steps: <https://youtu.be/DJKWW3ZLDds>

[Flags - Section 5] Run local Python app

1. On your computer, open a command prompt in the app directory: `flags/sample-app`

2. Update `server.py` with your model id and api key

3. Install app libraries:

```
> pip install -r requirements.txt
```

Mac users:

```
> pip install --user -r requirements.txt
```

4. Run `server.py`:

```
> python server.py
```

5. In a web browser, go to: <http://localhost:8000>

Video demonstrating these steps: <https://youtu.be/DsYfjF5PXnk>

[Flags - Section 6] Push app to IBM Cloud

1. Create a Python app starter in IBM Cloud: <https://console.bluemix.net/catalog/starters/python>
2. On your computer, open a command prompt in the app directory: `flags/sample-app`
3. In `manifest.yml` replace “app-name” with the name you chose for your starter app
4. Log in to IBM Cloud:

```
> bx login
```
5. Target the CloudFoundry API endpoint:

```
> bx target --cf
```
6. Push your app to IBM Cloud:

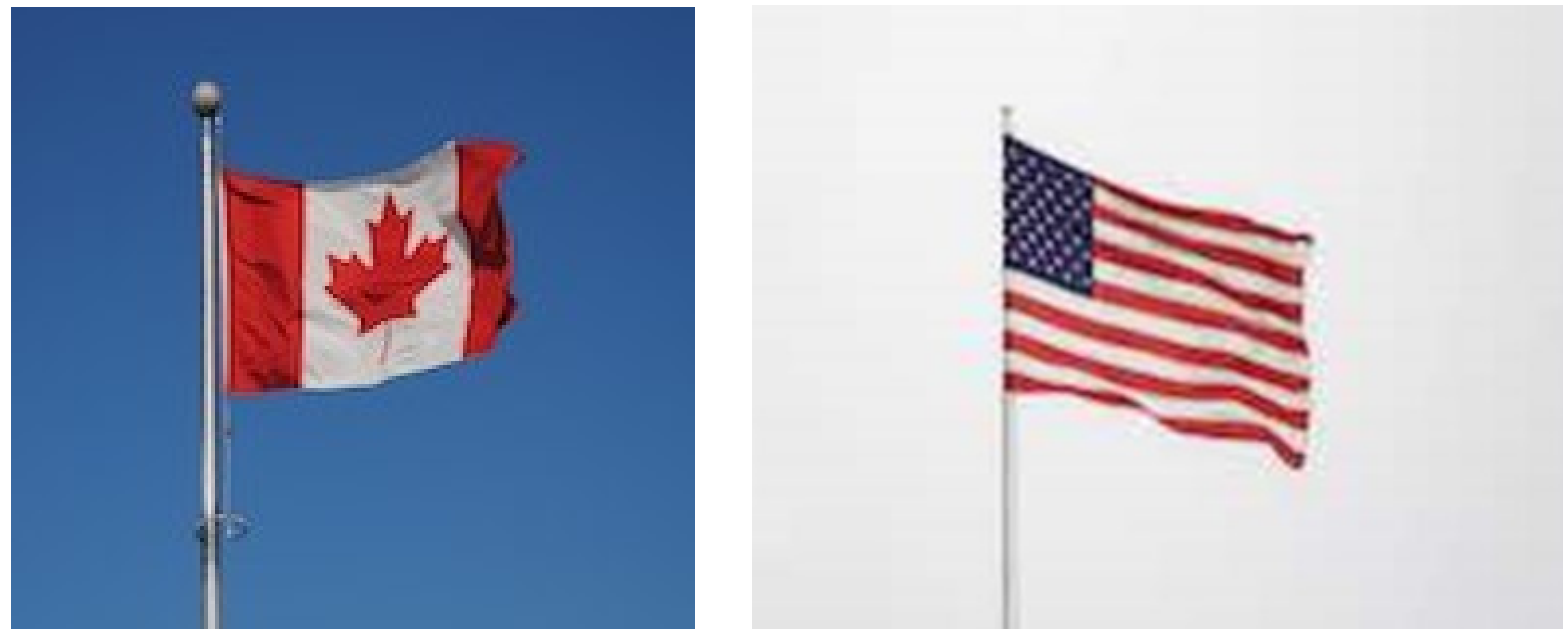
```
> bx app push
```
7. In a web browser, go to the url of your web app

Video demonstrating these steps: https://youtu.be/wepLdux_1og

Now: work on another sample on your own

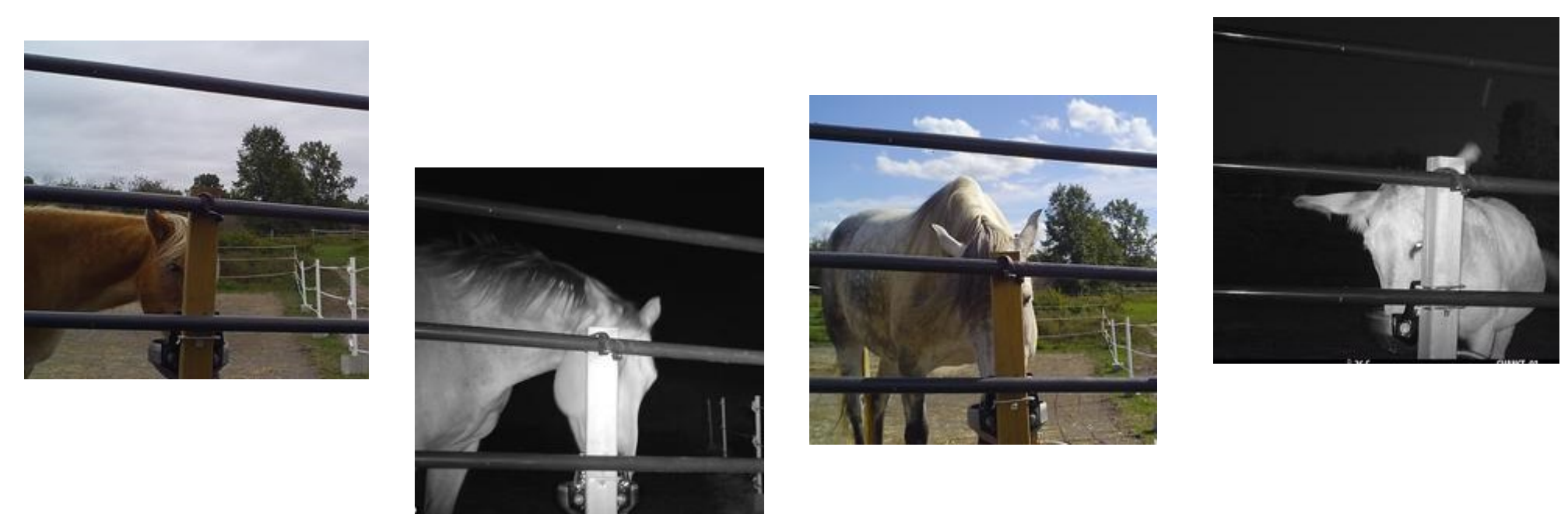
Flags

[\[link\]](#)



Water Consumption

[\[link\]](#)



Search & Rescue

[\[link\]](#)



Cat Translator

[\[link\]](#)

