Pragmatic ML development

with SciKit-Learn & TensorFlow using Google ML Engine

Google Cloud Platform

@ZackAkil

1. Why is your code not working on my machine?

2. How’s the model coming along?

3. How do we put your model in our app?

Why is your code not working on my machine?

Reproducing code

```
scikit-image==0.13.1
scikit-learn==0.19.2
scipy==0.19.1
seaborn==0.7.1
simplegeneric==0.8.1
six==1.11.0
spacy==2.0.12
statsmodels==0.8.0
sympy==1.1.1
tensorboard==1.10.0
tensorflow==1.11.0rc2
tensorflow-hub==0.1.1
```

requirements.txt

The meetup scenario (extreme reproducibility)

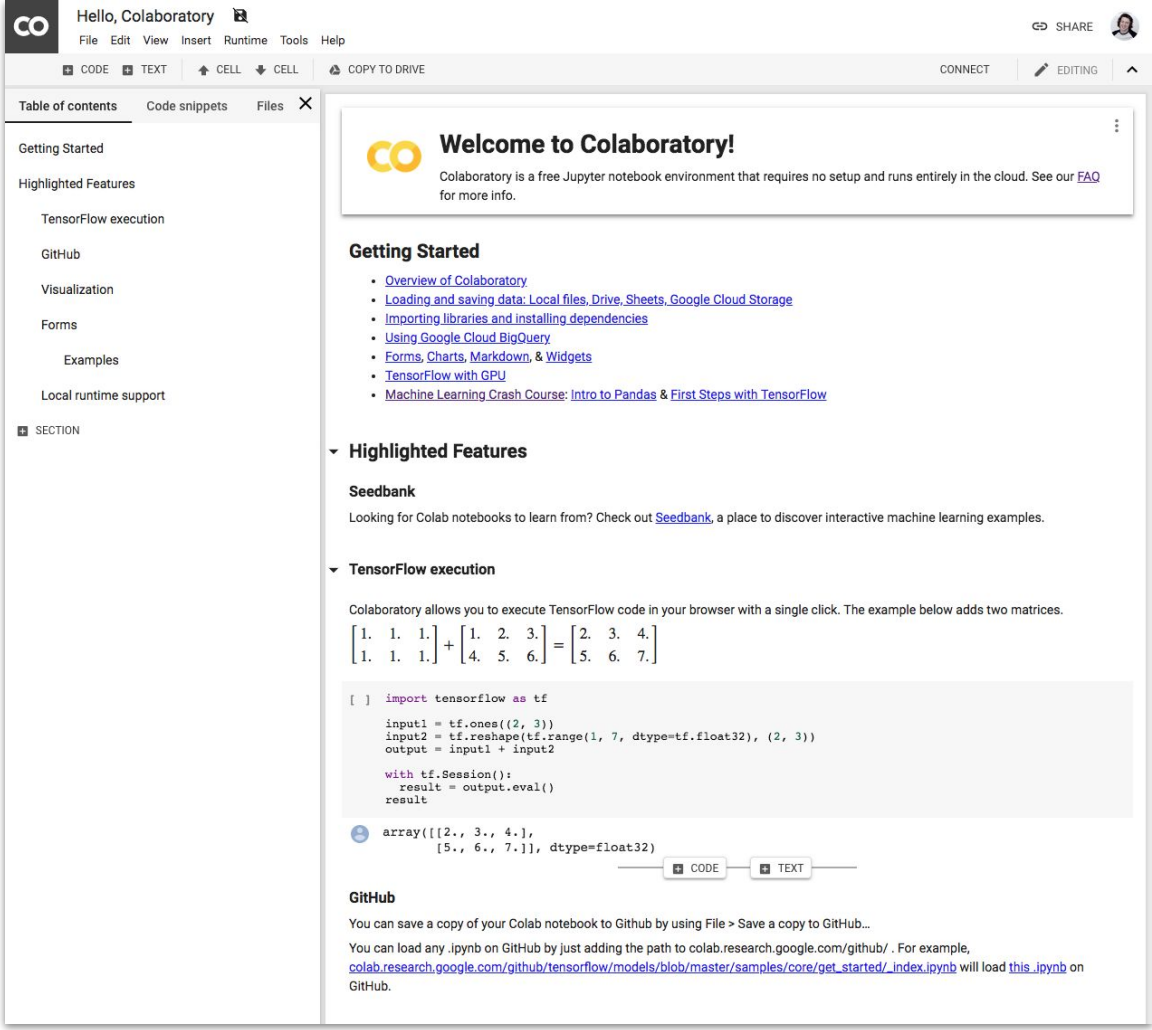
Local environment

kaggle

colab

colab.research.google.com

Collaborative Jupyter Notebooks



Colab Demo

Fix_my_code_buddy.ipynb

Working in a temporary environment

- Forces you to write more repeatable code
- Pushes you to remotely stored data
- Forces you to explicitly install non-standard libraries
  E.g this image processing notebook

Cloud Storage

How’s the model coming along?

Project approach

1. Is ML a want or a need?
2. Productionise every model
3. Stop when you have a “good enough” model

Want or need?

> If you don’t need ML,
>
> but you want to learn about it,
>
> then you need it.
>
> *(for educational purposes)*

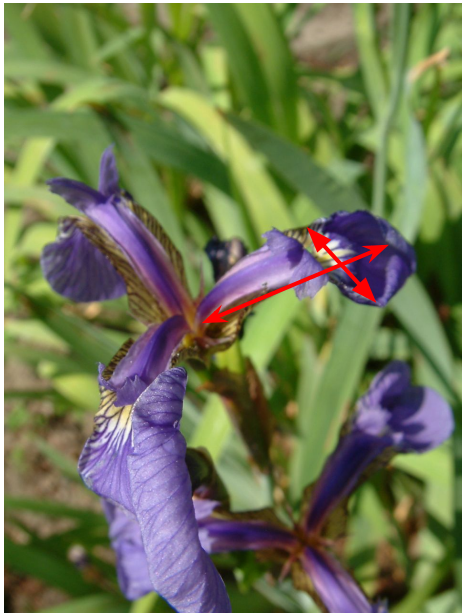
Deploy every model*

Cloud Functions

- Might already be good enough (*so you can stop there*)
- Gives the application developers something to work with
- Quickly confirms if the project is going in the right direction

*Especially the heuristic model

Not Iris dataset



Petal Length

Petal Width

Sepal Length

Sepal Width

not Setosa

not Versicolor

not Virginica

Cloud Functions demo

notebook

How do we put your model in our app?

Deploying every model with ML Engine

XGBoost

Cloud ML Engine

Keras*

Cloud ML Engine Demo

notebook

Final pragmatic advice

> **Manage expectations!**
>
> Machine Learning is mostly experimental,
>
> It’s not guaranteed to work.

Thank you

@ZackAkil

Google Cloud Platform