GENERAL ASSEMBLY

INTRO TO DATA SCIENCE

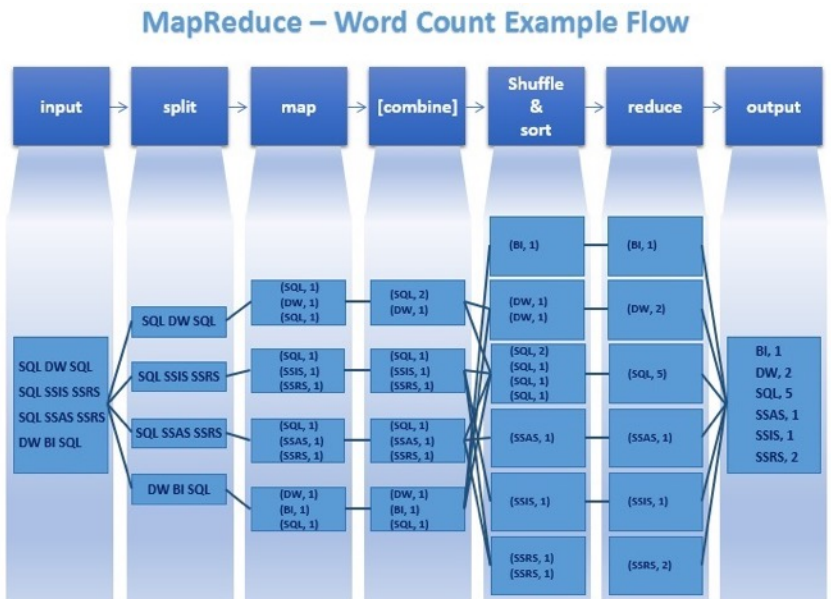
WEB DEVELOPMENT WITH FLASK/HEROKU

LAST TIME:

I. BIG DATA
II. PROGRAMMING MODEL
III. IMPLEMENTATION DETAILS
IV. WORD COUNT EXAMPLE

EXERCISE:

V. MAP-REDUCE USING PYTHON



QUESTIONS?

WHAT WAS THE MOST INTERESTING THING YOU LEARNT?

WHAT WAS THE HARDEST TO GRASP?

I. WHAT IS WEB DEVELOPMENT?
II. WHAT IS HEROKU / FLASK?
III. MVC
IV. DEPLOYING KNN

- UNDERSTAND MODEL - VIEW - CONTROLLER
- BE ABLE TO DEPLOY A MODEL ON HEROKU
- BE ABLE TO USE VIRTUAL ENVIRONMENTS

WEB DEVELOPMENT

Q: what is web development

A: The work involved with building and maintaining a live website

Two types of web development

Front-end:
HTML/CSS, Responsive design
Makes things pretty / easy to use

Back-end:
Many languages, Model View Controller, Databases
Makes the site "work"

Full-stack Development comprises of both front-end and back-end work

Part of being a web developer is knowing the technologies used:

Web-framework
database and site code


Deployment
"serve" the website
so that other people can use it

HEROKU / FLASK

Web Development is hard..

Which is why GA has several classes dedicated to it

We will use two very simple web development tools:
Heroku and **Flask**

Did someone say Flask!?

Did someone say Flask!?

Flask is a micro-web-framework based entirely in python

What does that mean?

It means we can write the entire backend in Python!

Did someone say Heroku!?

Heroku is a Salesforce company that lets us deploy our websites easily

What does that mean?

We use heroku to rent servers to host the website

One thing that stays constant over all technologies is the idea of the

Model View Controller paradigm

MODEL VIEW CONTROLLER

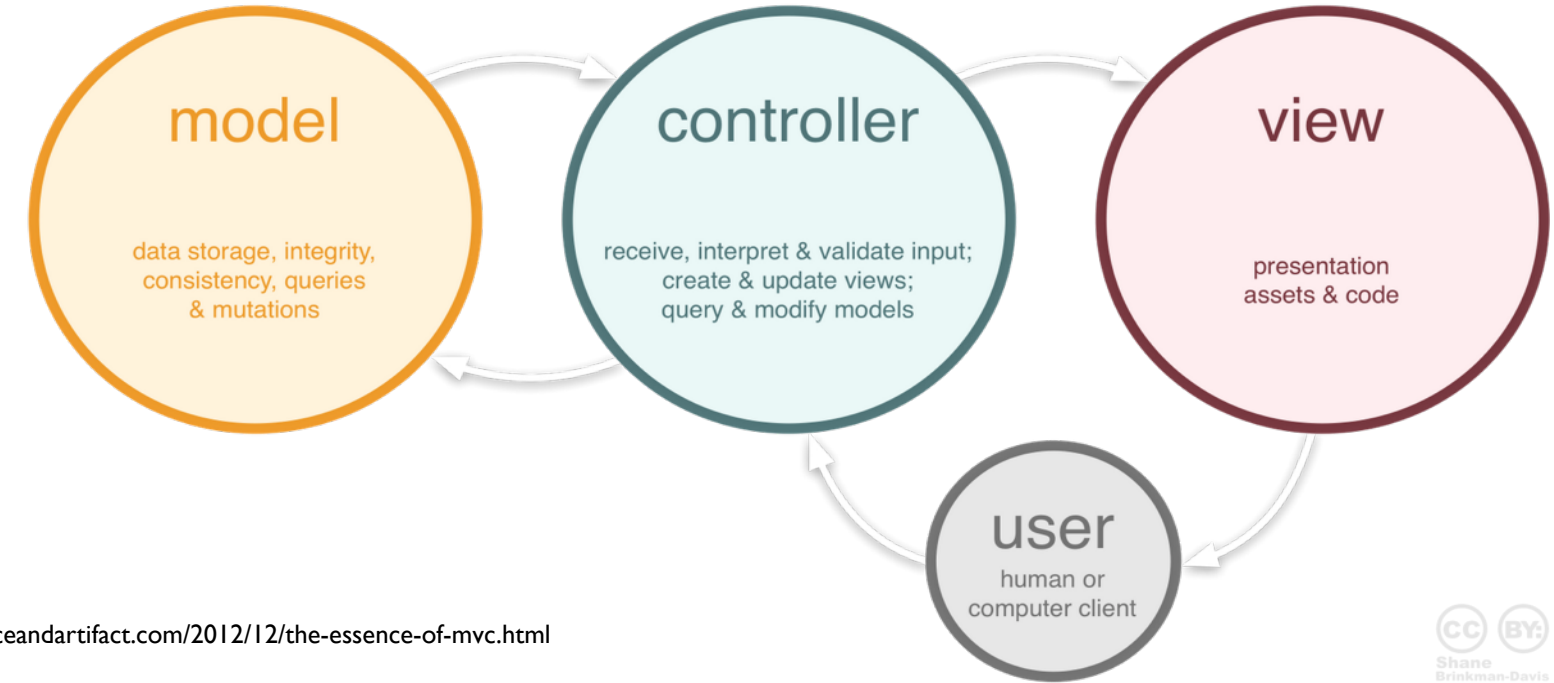
Model View Controller:

Model View Controller:

Is a way of life

Model View Controller:

But actually it's a software design pattern specifically for web apps



Source: http://www.essenceandartifact.com/2012/12/the-essence-of-mvc.html

CC BY: Shane Brinkman-Davis

Model View Controller:

Model
- Responsible for managing the data
- It's a database essentially!

View
- Presents the data / app
- Responsible for design / user experience

Controller
- Responds to user input and performs operations based on it
- Eg. User inputs a number of neighbors and the controller trains the model

Model View Controller:

Model
- Responsible for managing the data
- It's a database essentially!

View
- Presents the data / app
- Responsible for design / user experience

Controller
- Responds to user input and performs operations based on it
- Eg. User inputs a number of neighbors and the controller trains the model

QUESTION:

Which ones are handled by

Back end developers?

Front end?

Model View Controller:

Model (Backend)
- Responsible for managing the data
- It's a database essentially!

View (Frontend)
- Presents the data / app
- Responsible for design / user experience

Controller (Backend / Frontend)
- Responds to user input and performs operations based on it
- Eg. User inputs a number of neighbors and the controller trains the model

QUESTION:

Which ones are handled by Back end developers?

Front end?

DEPLOYING KNN

Sample Flask App

https://github.com/ghego/iris_calculator

Notice we have:

1. Models
2. Views (called templates)
3. Controller (controller.py)

Sample Flask App

https://github.com/ghego/iris_calculator

Go ahead and clone it

NOT IN YOUR OTHER GIT REPOSITORY ☺

Sample Flask App

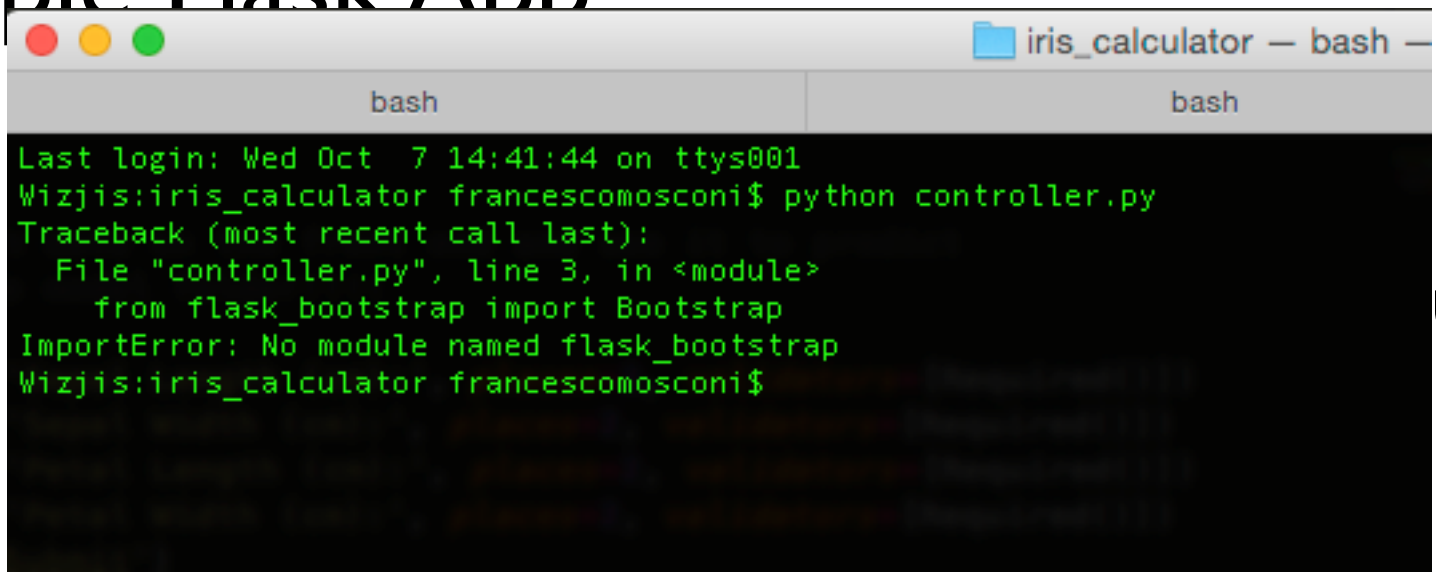
https://github.com/ghego/iris_calculator

To run locally, go to root and run

python controller.py

Go to **http://127.0.0.1:5000/**

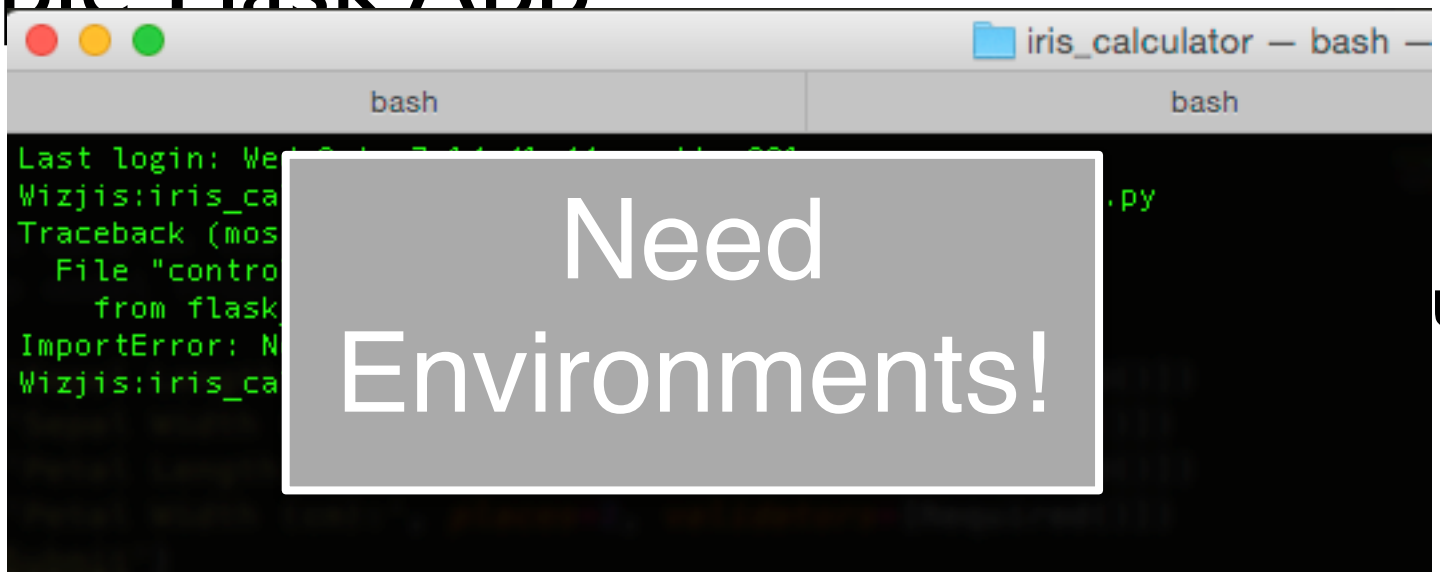
Sample Flask App



un

Go to **http://127.0.0.1:5000/**

Sample Flask App



un

Go to **http://127.0.0.1:5000/**

Aside: Python environments

A python project has dependencies
e.g.…

Aside: Python environments

A python project has dependencies
e.g.…
pandas
numpy
scikit-learn

Q: How does one keep track of dependencies and make sure that a program runs on a different platform?

Q: How does one keep track of dependencies and make sure that a program runs on a different platform?

A: Environments!

Q: How does one keep track of dependencies and make sure that a program runs on a different platform?

A: Environments!

Go ahead and read here:

http://conda.pydata.org/docs/using/envs.html

http://pip.readthedocs.org/en/stable/reference/pip_freeze/

How do I create a new clean environment that only contains the following packages:

- python
- numpy1.8.1
- scikit-learn0.15.2
- scipy0.14.0

How do I create a new clean environment that only contains the following packages:

- python
- numpy1.8.1
- scikit-learn0.15.2
- scipy0.14.0

```
conda create -n test_env python numpy=1.8.1 scikit-learn=0.15.2 scipy=0.14.0
```

How do I activate the environment?

How do I de-activate the environment?

How do I delete the environment?

How do I activate the environment?

```
source activate test_env
```

How do I de-activate the environment?

```
source deactivate
```

How do I delete the environment?

```
conda remove -n test_env --all
```

Let's get back: Sample Flask App

https://github.com/ghego/iris_calculator

NOTE:

You may not have the required modules to run it right now..

If not, run

pip install –r requirements_clean.txt

To run locally, go to root and run

python controller.py

Go to **http://127.0.0.1:5000/**

We have two forms

The top form **trains the model**

The bottom form **predicts incoming data**

When we submit the data….

We have two forms

The top form **trains the model**

The bottom form **predicts incoming data**

When we submit the data….

QUESTION:

Which part of MVC handles the input?

We have two forms

The top form **trains the model**

The bottom form **predicts incoming data**

When we submit the data the **controller** handles it!

Iris Calculator

Please fill out the form below!

Number of Neighbors: 2
Paramter 2 (Optional)
Paramter 3 (Optional)
Paramter 4 (Optional)

Submit

Sepal Length (cm):
Sepal Width (cm):
Petal Length (cm):
Petal Width (cm):

Submit

Note:

Note we included 3 optional parameters in case you need to train on more than just one!

The machine learning model lives in the **model** folder
not to be confused with the model in MVC

It is pickled…

The machine learning model lives in the **model** folder
not to be confused with the model in MVC

It is pickled…

You know, the standard mechanism
for serializing an object.

Essentially we can transform a
python object into a file!

You can pickle anything!!

1. sklearn models
2. Jsons!
3. Strings!
4. Your own models

Comprehensive Step by Step

1. Sign up for Heroku at: http://heroku.com
2. Create a new app (make sure Heroku toolbelt is installed)
3. Clone our flask app https://github.com/ghego/iris_calculator
    1. Change and test at will
4. Run the command: **heroku git:remote –a <APP>**
5. Install the custom build back for scipy and numpy
    1. heroku config:set BUILDPACK_URL=https://github.com/thenovices/heroku-buildpack-scipy –app <APP>
    2. Run the command above in the root of your app (with the toolbelt installed)
6. Use as normal git repository:
    1. Git add, commit, etc...
    2. Git push heroku master (instead of origin)
7. Amaze people with your live KNN

1. SIGN UP FOR HEROKU

Self-explanatory?

http://heroku.com

2. CREATE A NEW APP (MAKE SURE HEROKU TOOLBELT IS INSTALLED)

Not self-explanatory

https://toolbelt.heroku.com/

Type into your console:

heroku login

3. CLONE OUR FLASK APP
HTTPS://GITHUB.COM/GHEGO/IRIS_CALCULATOR

Self-explanatory?

Now you can run it locally!! (remember run python controller.py)

4. RUN THE COMMAND: HEROKU CREATE

At the root of the directory

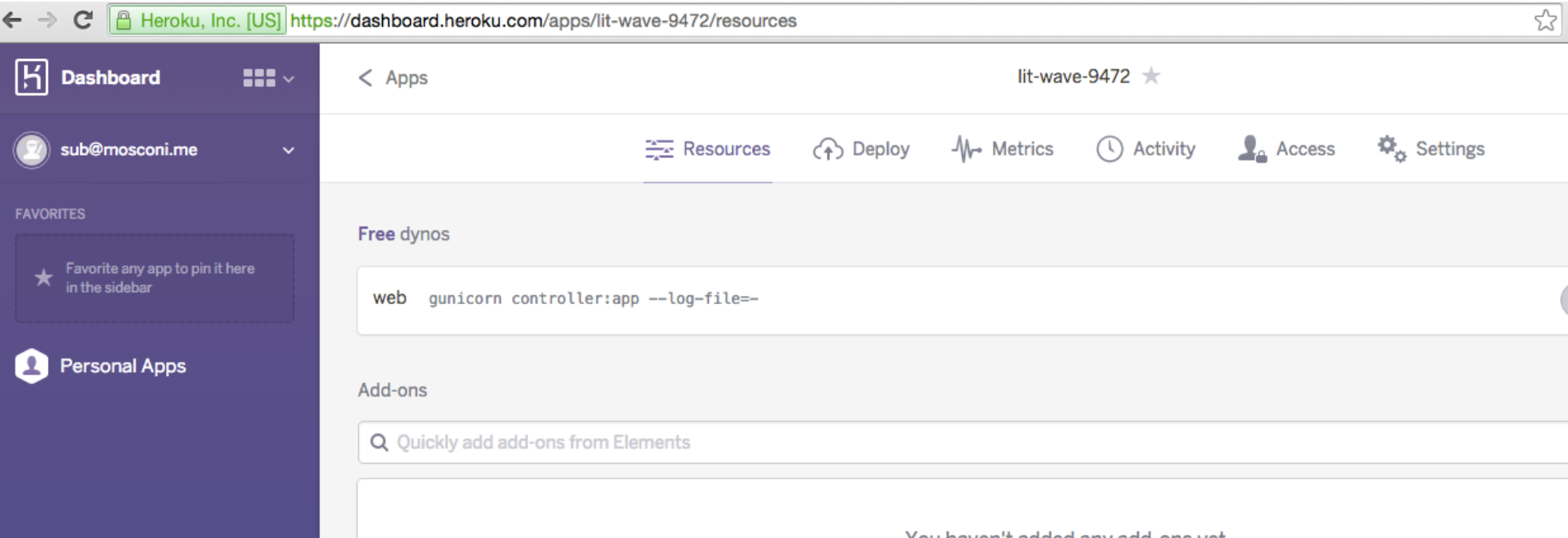
This adds a new git "remote"

Essentially a new place to push ☺

Check this by running *git remote –v*

You should see origin and heroku

4. RUN THE COMMAND: HEROKU CREATE



5. INSTALL THE CUSTOM BUILD BACK FOR SCIPY AND NUMPY

Not self-explanatory

https://github.com/thenovices/heroku-buildpack-scipy

Run:

heroku config:set BUILDPACK_URL=https://github.com/thenovices/heroku-buildpack-scipy - -app <APP>

At the root of the directory

6. USE AS NORMAL GIT REPOSITORY

Self-explanatory?

git add .

git commit –m “I am a genius”

Git push heroku master

6. USE AS NORMAL GIT REPOSITORY

Self-explanatory?

git add .

git commit –m “I am a genius”

Git push heroku master

Note:

It is installing a bunch of modules because of the requirements.txt file

7. AMAZE PEOPLE WITH YOUR LIVE KNN

Self-explanatory!!!!!

Note:

Your unique website will have your app name instead of fm-iris

https://fm-iris.herokuapp.com/

WHAT NOW?!!?!?!!1?

Put in your own machine learning model!

1. **Build your model else where**
2. **Load it into the model folder manually**