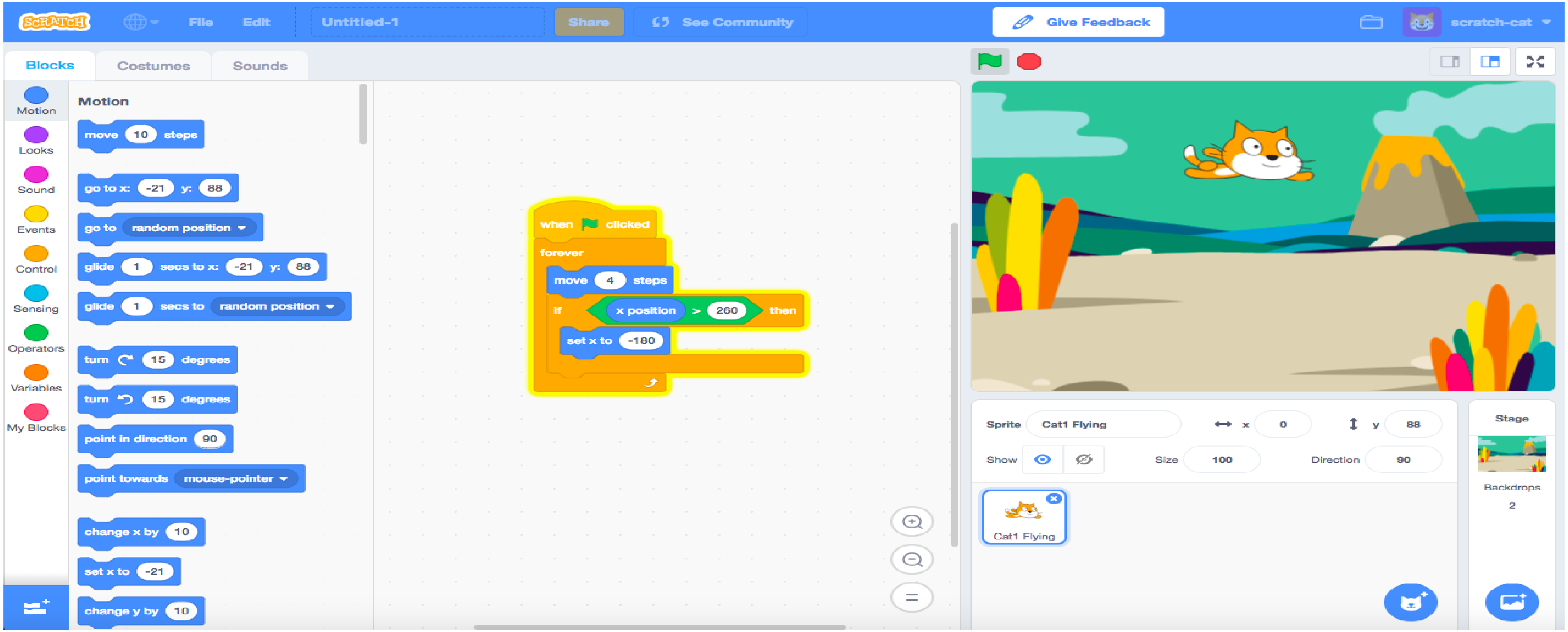
SCRATCH DATA ANALYSIS

BY

MATT DAVIDSON, JACOB COHEN, FAISAL ALSALLUM, AND CRYSTAL YU

BACKGROUND:

- Scratch is a block-based visual programming language
- Can be used to program a user own interactive stories, games, and animations
- It is a project of the Lifelong Kindergarten Group at the MIT Media Lab.
- It is designed especially for ages 8 to 16, but is used by people of all ages.
- The projects are scored by scratch own scoring system



```
when ⚑ clicked
forever
    move 4 steps
    if x position > 260 then
        set x to -180
```

Sprite Cat1 Flying ↔ x 0 ↕ y 88

Show Size 100 Direction 90

Stage

Backdrops 2

EXAMPLE

ABOUT THE DATA SET:

- 250K Scratch projects from 100K different authors scraped from the Scratch project repository.
- Projects’ source code and metadata were encoded into a database.
- Projects were evaluated by the dataset providers in terms of programming skills and mastery and scoring the results.

GOALS:

- Create a web app tool for a user to predict the score of a given certain inputs, using the data set as a training data.
- Search for example of projects given a key word.

PYTHON WEB PACKAGES

DASH WEB PACKAGE:

- Designed for interactive visualization of data analytics
- 1:1.5 ratio of open to closed issues on github; active community
- Easy to use with only python code

How it works:

- Builds off Flask, Plotly, and React to create a pretty web interface without detailed backend development

BOKEH WEB PACKAGE:

- A visualization library that incorporates plots, tools, and widgets to explore data
- Works well with large datasets
- Can create simple static graphics as well as complex interactive tools using only python code
- Active github community, with over 5000 closed issues

How it works:

- Graphics and tools are built in layers
- Submodules are used to incorporate different functionalities

FLASK WEB PACKAGE:

- Micro-framework: very few dependencies or defaults
- Can create a simple webpage in handful of lines of code
- Have to make choices about every aspect of the backend (e.g. database, user inputs) and frontend (e.g. html/css formatting)
- Active github presence, nearly 2000 closed issues

How it works:

- includes a decorator that turns Python functions into http response at a specific endpoint (e.g. 127.0.0.0:5000/data)
- routes can be static, or generated dynamically based on the inputs
- server can be run locally or hosted (e.g. on AWS, Azure)

PROS AND CONS

Dash byplotly

Pros:
- Builds off Flask, Plotly, and React
- Easier to implement than Bokeh and Flask
- Many available online resources

Cons:
- Missing some of the other packages' more advanced features

bokeh

Pros:
- Well documented, with many examples from different fields
- Works in Jupyter notebook

Cons:
- Need to separately import submodules for different functionalities

Flask

Pros:
- easy to deploy simple pages
- convert python functions to pages

Cons:

-lots of design decisions

-benefits from knowledge of other languages (e.g. html, javascript)

THANK YOU