Image Features

How was AI4All?

The year was 2018. We arrived at 8:32am on Monday, June 25th, passing through the east entrance of Stanford campus. When we left the bus, I discovered that the temperature was an uncharacteristically chilly 57 degrees, so I put on my blue jacket and wore it for between 10 and 12 minutes until we arrived at Cubberley Auditorium between the library and campus main quad. Lessons started shortly thereafter, but I was distracted by a fly buzzing around the window. I believe the glass manufacturer was ...

This story *technically* contains a lot of information about AI4All.

But it’s not the information that matters! Your mom cares about:

- The friends you made.
- How you felt being there.
- Topics you learned about.
- If you were well fed.

Feature Extraction

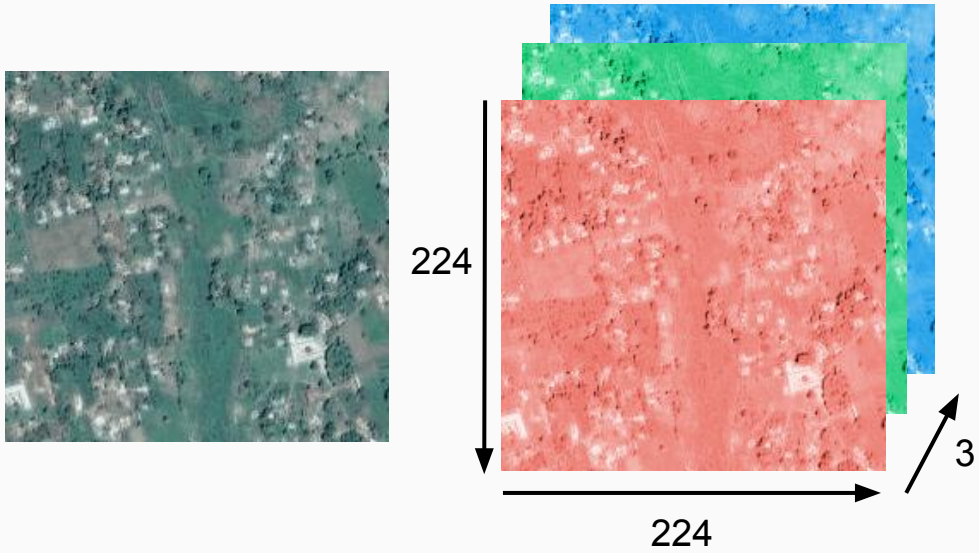
This story *technically* contains a lot of information about AI4All.

But it's not the information that matters! Your mom cares about:

- The friends you made.
- Topics you learned about.
- How you felt being there.
- If you were well fed.

FEATURES describe events, people, or objects, by throwing out unnecessary information to focus on things that matter.

Feature Extraction



Coming up with good **Features** for images is *extremely* important, because images are so large.

The satellite images we downloaded yesterday are each 224x224x3 arrays…

That's **163,968** numbers needed to describe just one picture!

Can you think of some features that could help us identify poverty?

Feature Extraction

A Feature Vector is an array of values that can summarize items in our dataset.

We say a feature is **predictive** when it's useful for telling us what we want to know.

Population	Crop Yield	Deforestation	Income
25,000	0.5	0.08	4,500

Feature Extraction

25,000	0.50	0.08	4,500

10,000	0.32	0.10	3,000

7,000	0.26	0.18	500

52,000	0.78	0.10	6,700

If we can extract a set of features for our dataset, we can greatly reduce the number of values we store.

For example, using **4 features** we can store data for **650** locations in a matrix with shape **(650, 4)**.

That's much smaller than a matrix of shape **(650, 163,968)**

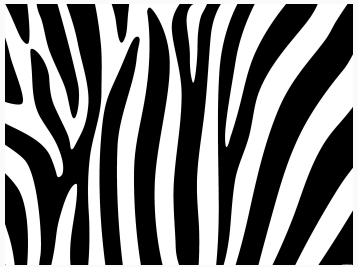
Can you think of a problem with using “Population”, “Crop Yield”, “Deforestation” and “Income” as features?

Image Features

Features must be extracted **directly** from our data.

Things like *shapes*, *patterns*, *colors*.

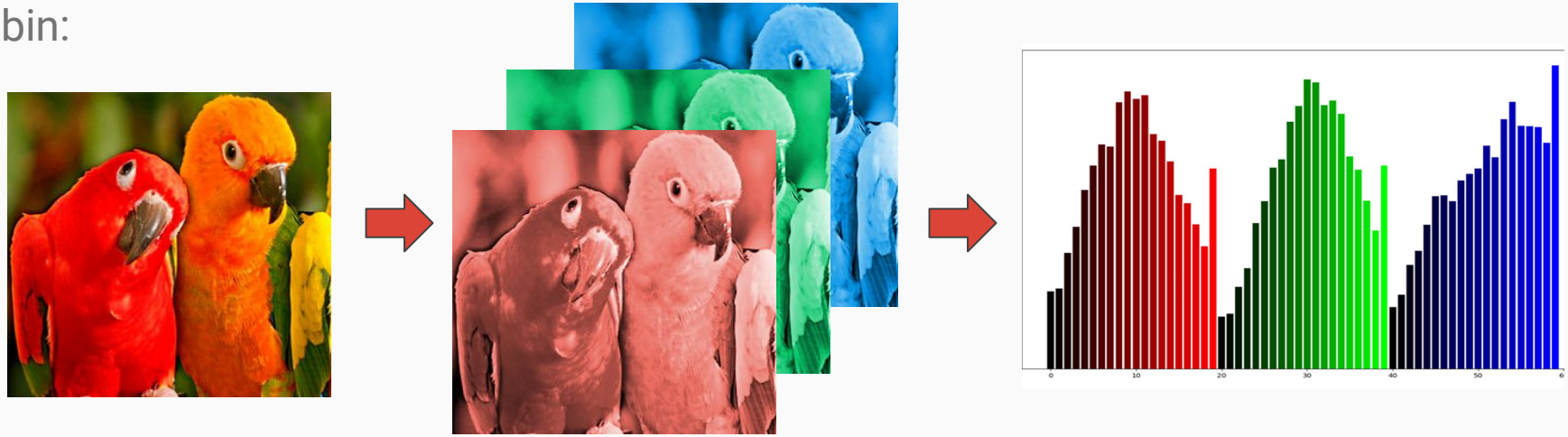
And all of them must be defined in terms of **algorithms** and **mathematics**, so we can implement them!



We now understand colors pretty well. How can we aggregate colors into features?

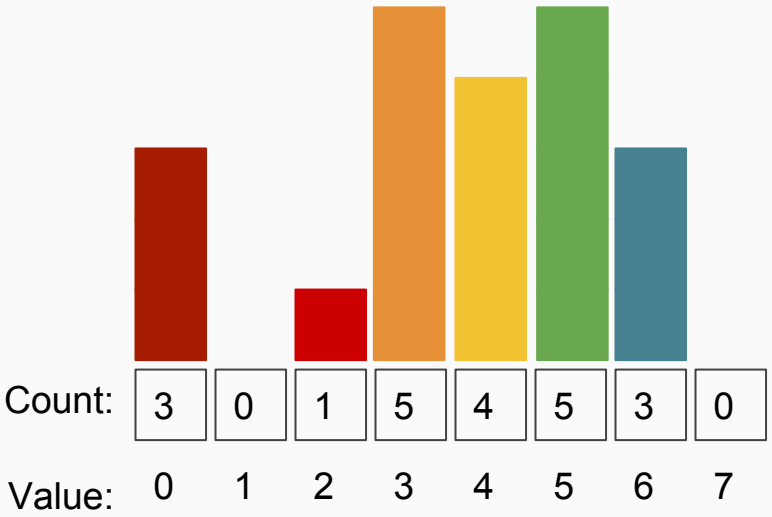
Image Features

Color Histograms go over every pixel in an image and store its intensity in a bin:



This transforms our **3D array** into a simple, one-dimensional list.

Histograms

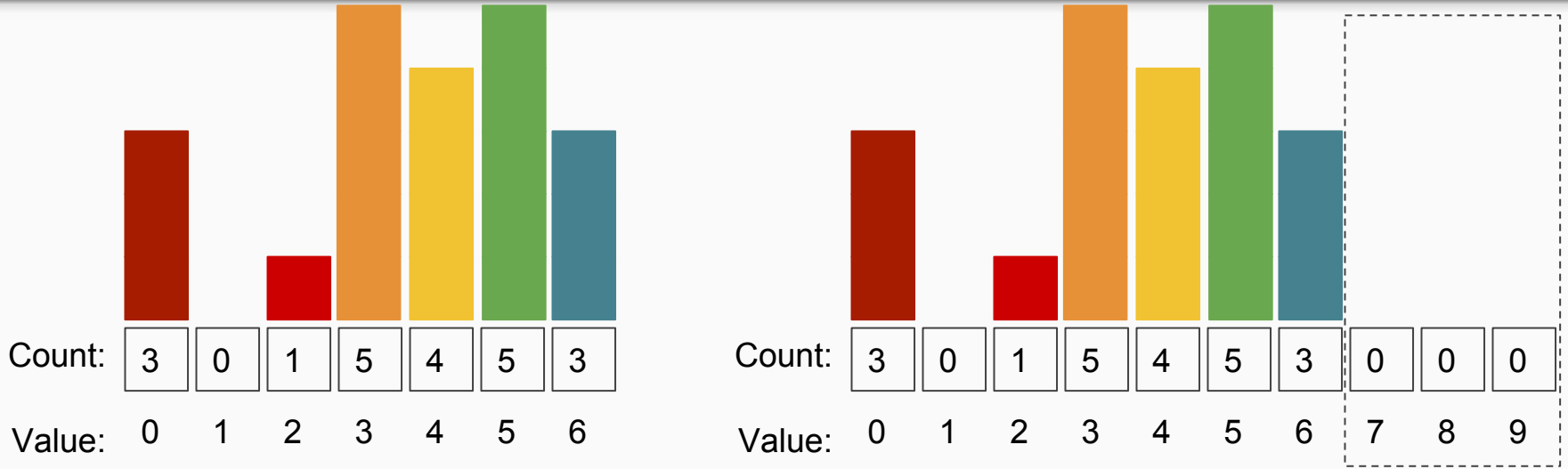


Histograms store a **count** for a set of **values** in a matrix with one axis (a 1D array)

For example, we can read this histogram as saying that there are:

3 “zeroes”, 0 “ones”, 1 “twos”, etc ...

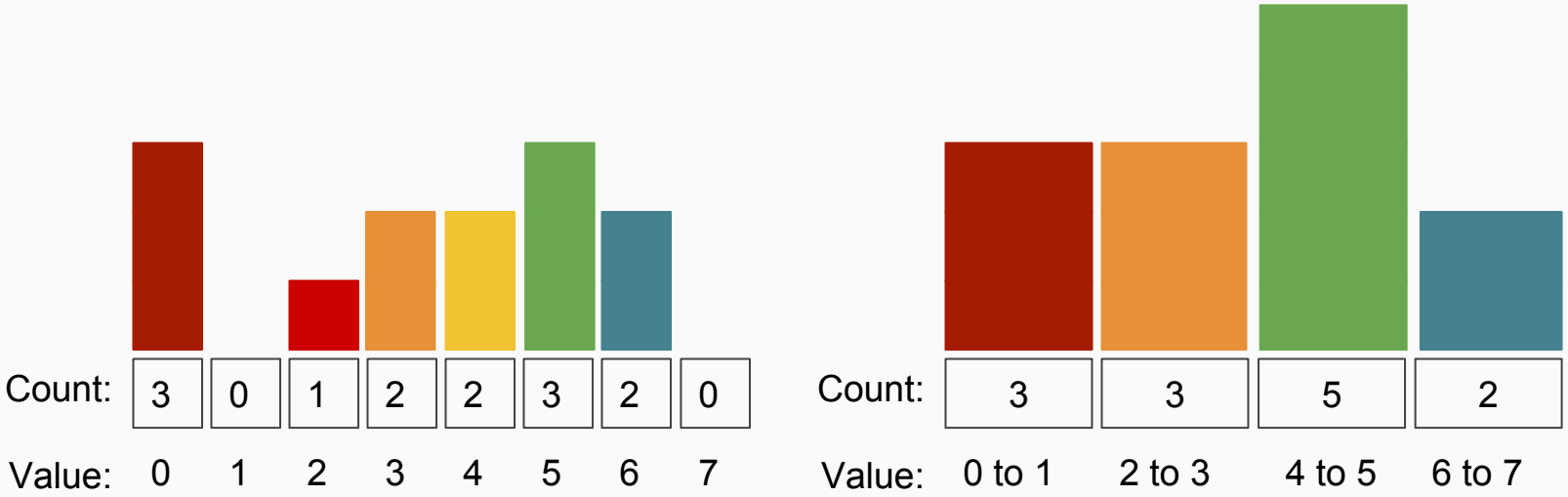
Histograms: Range



You must specify a **range** telling the histogram what sort of **values** to expect.

Left: The range is (0, 6). **Right:** The range is (0, 9).

Histograms: Bins



We can also choose the number of **Bins**: the less you have the more values are grouped.

Left: has *8 bins*. **Right:** has *4 bins*.

Histograms: NumPy Command

h, *edges* = *np.histogram(…)*

The Python package NumPy (np) provides a histogram implementation.

Return Values:

h: Is the histogram itself. Or a 1D-array with shape: *(bins,)*

edges: A list of cut-off points between bins, such as [0,2,4,6]

Extra Credit Question! Can we make Color Histograms out of HSV images too?

color_features.ipynb

1. Implement the color histogram function *“image_to_histogram”*
2. Run a simple machine learning experiment to classify **trucks** versus **planes**.
3. View the *mean histogram* for **Trucks** and **Planes**.
4. Run the same experiment on Satellite data.