

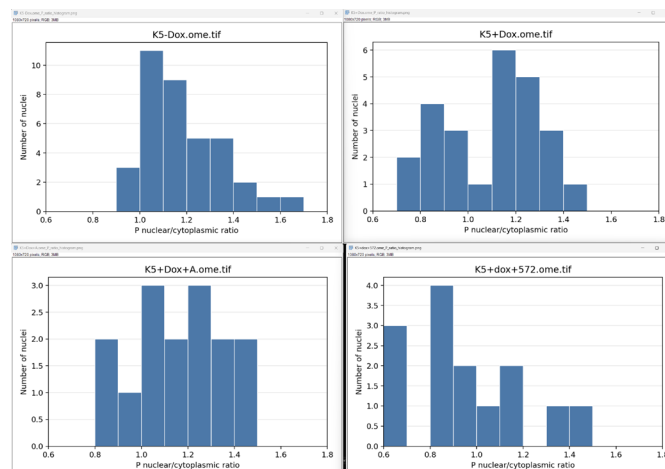
## BP 210: Biological Light Microscopy

### Lab 10: Use Python and generative AI for image analysis

The idea is to bring your own images and use generative AI (codex or Claude code) to write and execute the analysis for you.

As an example (and to get you started), I included 4 images that a researcher gave me with the desire to get cytoplasmic ratio of a marker. The images have a nuclear channel and a channel with the marker. Just to give you an idea, here are the prompts that I used (including typos) with the resulting histogram. The AI (codex) gave me a Python script that I can run again later.

- I like to analyze the images in `example_data/NuclearCytoplasmicRatio`. These images contain two channels, the first is a nuclear marker, the second is called "P". I would like to get a list for each image with the number of nuclei, and for each nucleus/cell, the ratio of the concentration of P in the nucleus and cytoplasm. I would like you to use the `cellquant` code in this directory and make sure to generate masks for nuclei (using the nuclear marker) and cytoplasm (using the P signal) using `Cellpose`. Also, please store the images of the masks for later inspection.
- Some nuclei/cytoplasm are clearly too large and made up of multiple cells. Are these somehow excluded from analysis?
- Can you make that better fix and rerun?
- Can you add the centroids of the nuclei to the table so that I can refind them in the images?
- Could you write the nucleus ID in the nuclear mask images?
- Looks reasonable. Can you make histograms of the number of nuclei with given ranges of nuclear/cytoplasmic ratio of P for each of the 4 images/conditions?



To play more with the Cellquant script itself (note the slashes and backslashes):

```
uv run cellquant.py example_data/mammalian_SGs/ \
  "1:DAPI:nucleus" "2:G3BP1:quantify" "3:PABPC1:quantify" \
  --cell-type mammalian \
  --out example_data/mammalian_SGs/my_output/ \
  --filename-pattern "MAX_{condition}_rep{replicate}"
```

Check the QC overlays in my\_output/qc/

Find your data in my\_output/cells.csv