

Which cat does this compound look like?

Group: cytohackR

Christian, Marie, Joerg, Shantanu, Becka, Sebastian

Extract images from IDR

The screenshot displays the Webclient interface for cytohack.openmicroscopy.org. The browser address bar shows the URL `cytohack.openmicroscopy.org/webclient/?show=plate-4403`. The interface is divided into several sections:

- Top Navigation:** Includes tabs for IDR, Studies, Genes, Phenotypes, Cell Lines, siRNAs, Antibodies, Compounds, Organisms, and Help. A search bar and a user profile icon (Public User) are also present.
- Left Sidebar:** Features a file explorer under the 'demo Demo User' section. It lists various folders and files, including 'idr0001-graml-sysgro/screenA 192', 'idr0002-heriche-condensation/screenA 12', and '20585'.
- Main Content Area:** Displays a grid of microscopy images under the heading 'Index: Field#1'. The grid is organized into rows (A-I) and columns (1-10). Below the grid, there is a 'Wrap' checkbox and a slider for 'images per row'.
- Right Sidebar:** Contains a 'General' tab with details for plate '20585'. It includes fields for 'Plate ID: 4403', 'Owner: Demo User', 'Creation Date: 2016-05-19 06:00:39', and sections for 'Attributes', 'Attachments', 'Comments', 'Tags', 'Ratings', and 'Others'.

Want images? Easy D

- Query using Python API
- Create list of files
- Use imagedata/download

```
In [4]: omit = "uod/idr/incoming/idr0016-wawer-bioactivecompoundprofiling/"
        well = "_A01_"
        field = "_s1_"
        for x in list_files():
            if "Hoechst" in x[0] and field in x[0]:
                print x[0][len(omit):]
```

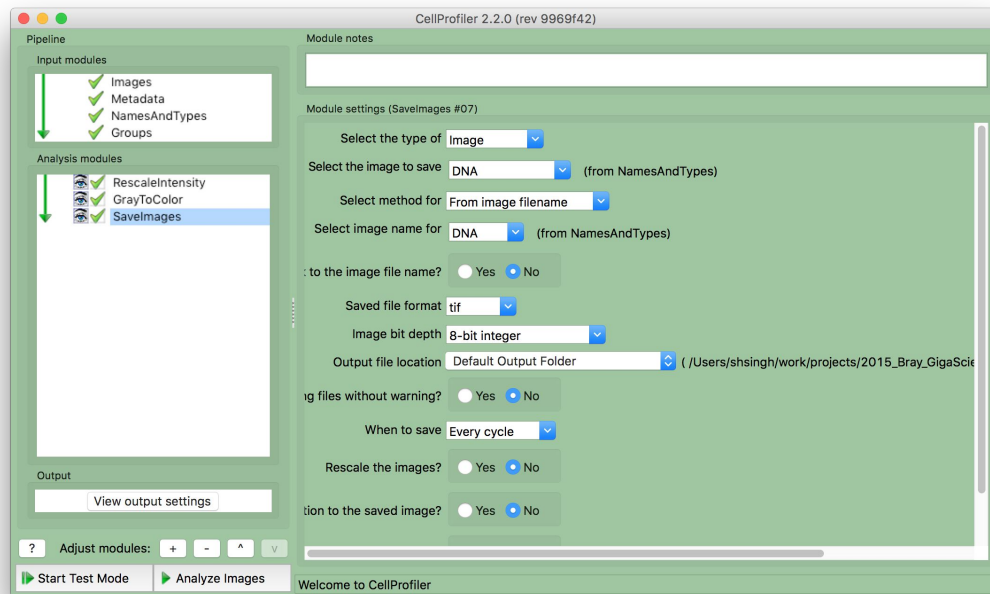
Connected to IDR...

```
2016-01-19-screens-bbbc022/20585-Hoechst/IXMtest_A01_s1_w164FBEEF7-F77C-4892-86F5-72D0160D4FB2.tif
2016-01-19-screens-bbbc022/20585-Hoechst/IXMtest_A02_s1_w17C9F8BDB-79F0-4F8B-852B-71161631F236.tif
2016-01-19-screens-bbbc022/20585-Hoechst/IXMtest_A03_s1_w1073D64AC-9B57-43E1-A34E-1C7C986DEEFA.tif
2016-01-19-screens-bbbc022/20585-Hoechst/IXMtest_A04_s1_w195151D27-9F95-44DE-A514-2ED183E548C9.tif
```

```
while read x; do docker run --rm -v
/tmp/cytohackR:/data imagedata/download idr0016 $x
/data/; done < ~/Desktop/DNA.txt
```

CellProfiler to convert images

- Read in all images
- Rescale to max intensity
- Make pseudo RGB



Extract features

- Use pre-trained Inception-BN network
- Use input to fully connected layer as features



Archive

Build Online Image Classification Service with Shiny and MXNetR

Dec 8, 2015 • Qiang Kou

Early this week, Google announced its [Cloud Vision API](#), which can detect the content of an image.

With the power of R and MXNet, you can try something very similar on your own laptop: an image classification shiny app. Thanks to the powerful shiny framework, it is implemented with no more than 150 lines of R code.

Image Classification using MXNetR

Let's do this!

Image

Upload Image

Use the URL

Upload a PNG / JPEG File:

Choose File

No file chosen



Get metadata

- Fetched mechanism by querying IDR

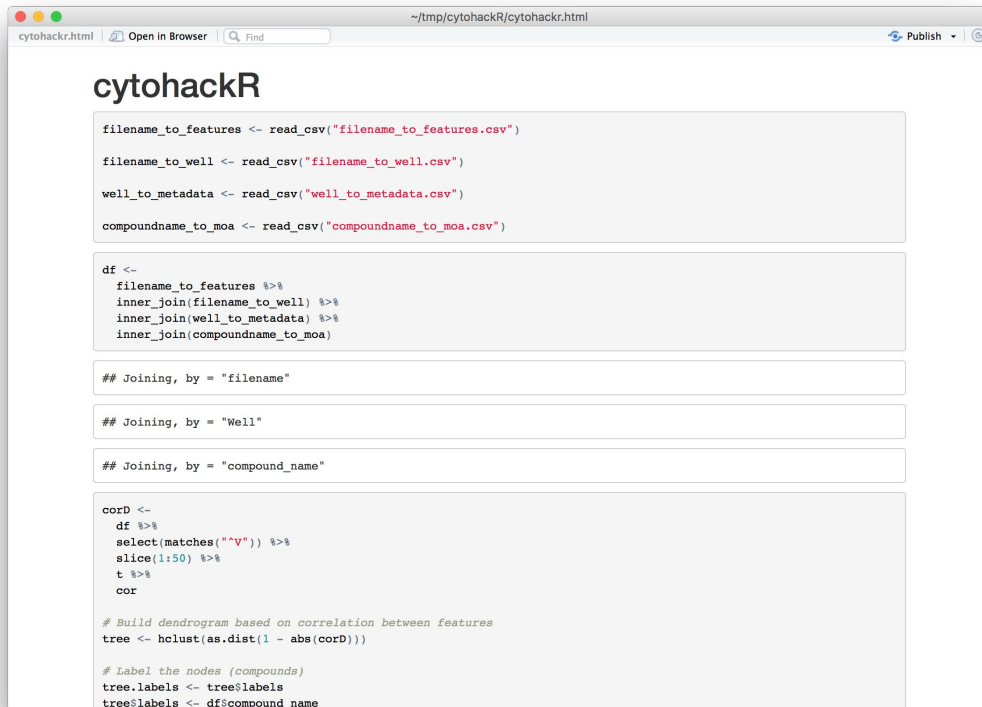
 query.py

```
1 import omero
2 compounds = unwrap(q.projection(("select distinct mv1.value, mv2.value from Image i "
3                                "join i.annotationLinks l1 join l1.child as a1 join a1.mapValue as mv1 "
4                                "join i.annotationLinks l2 join l2.child as a2 join a2.mapValue as mv2 "
5                                "join i.wellSamples ws join ws.well w join w.plate p "
6                                "where p.id = 4403 and mv1.name = 'Product Use Annotation' and mv2.name = 'Compound Name'")), None))
```

	Well	compound_name	mechanism
1	A01	NIFLUMIC ACID	analgesic
2	A03	NICARDIPINE	vasodilator
3	A04	6-Furfurylaminopurine	auxin
4	A06	Suxibuzone	analgesic
5	A07	LIDOCAINE	anesthetic
6	A11	GLYBURIDE	vasoconstrictor
7	A12	Mepenzolate bromide	cns agent
8	A17	FLUNARIZINE	neuroprotective agent
9	A19	FLUFENAMIC ACID	analgesic

Cluster compounds

- All data was exported as CSVs
- Read into R and analyzed
 - Use tidyverse!



The screenshot shows a web browser window with the address bar displaying `~/tmp/cytohackR/cytohackr.html`. The page title is `cytohackr.html`. The main content area is titled **cytohackR** and contains an R script editor. The script reads four CSV files, joins them into a single data frame, calculates a correlation matrix, and builds a dendrogram. The script is as follows:

```
filename_to_features <- read_csv("filename_to_features.csv")
filename_to_well <- read_csv("filename_to_well.csv")
well_to_metadata <- read_csv("well_to_metadata.csv")
compoundname_to_moa <- read_csv("compoundname_to_moa.csv")

df <-
  filename_to_features %>%
  inner_join(filename_to_well) %>%
  inner_join(well_to_metadata) %>%
  inner_join(compoundname_to_moa)

## Joining, by = "filename"

## Joining, by = "Well"

## Joining, by = "compound_name"

corD <-
  df %>%
  select(matches("^V")) %>%
  slice(1:50) %>%
  t %>%
  cor

# Build dendrogram based on correlation between features
tree <- hclust(as.dist(1 - abs(corD)))

# Label the nodes (compounds)
tree.labels <- tree$labels
tree$labels <- df$compound_name
```

Cluster compounds

