Project

Xiang Zhou

BIOSTAT 626

Data prepared by Jade Wang

Outline of the Project

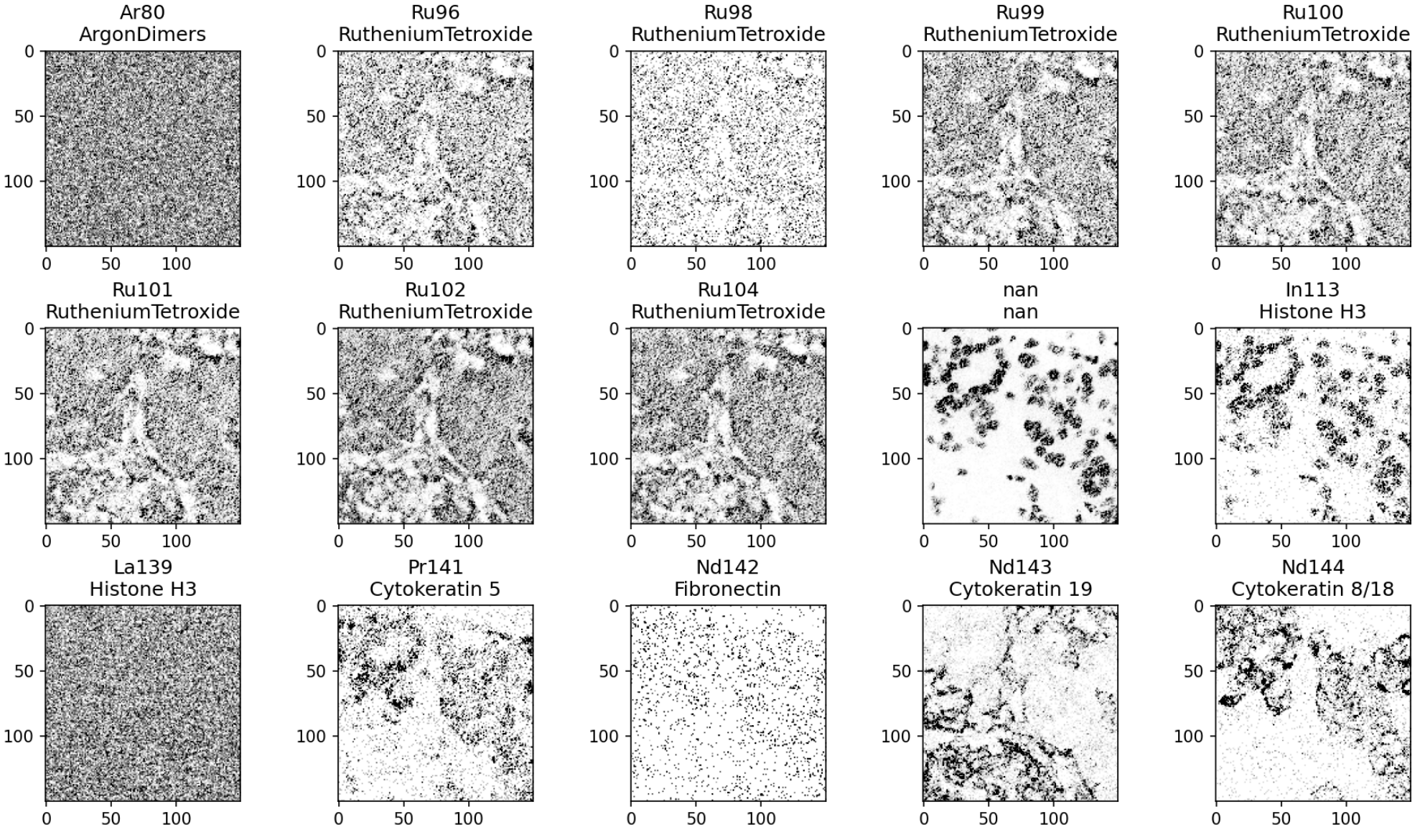
- Work on the project in teams.
- Each team consists of 1, 2, or 3 students.
- Please form your team and find your partner(s).
- Deadline: April 11th, Thursday.

Description

- The goal is to use cellular/subcellular resolution spatial proteomics data to predict cancer patient survival time.
- Outcome (continuous): patient survival time, in months.
- Features: one image per patient. Each image contains 52 channels, measuring the spatial expression pattern of 52 proteins on the tumor tissue.
- Training data: outcome and image data for 225 patients.
- Test data: image data for 56 patients.

Image Data

One sample, showing 15 out of 52 proteins



Dictionary File: Protein Names

index	Metal Tag	Target
protein1	Ar80	ArgonDimers
protein2	Ru96	RutheniumTetroxide
protein3	Ru98	RutheniumTetroxide
protein4	Ru99	RutheniumTetroxide
protein5	Ru100	RutheniumTetroxide
protein6	Ru101	RutheniumTetroxide
protein7	Ru102	RutheniumTetroxide
protein8	Ru104	RutheniumTetroxide
protein9		
protein10	In113	Histone H3
protein11	La139	Histone H3
protein12	Pr141	Cytokeratin 5
protein13	Nd142	Fibronectin
protein14	Nd143	Cytokeratin 19

Survival Outcome

Training data: n=225
Test data: n=56

id	OSmonth
1	80
2	73
3	59
4	53
5	169
6	171
7	33
8	66
9	54
10	107
11	102
12	63
13	35
14	72
15	51
16	95
17	70
18	68
19	78
20	18

Extract Features

- A sample python code is available to extract the average intensity measurement of each protein on each image.
- Therefore, 52 features are immediately available for each patient. This is a good starting point.
- Other possible features: intensity measure of cells that express the protein; cell size, shape; other features extracted from existing neural networks targeted for image analysis, etc.

Sample Python Code

```
# this is a demo code for computing the average intensity of
proteins of all images (save as .csv)
import numpy as np
import pandas as pd
import imageio
from IPython.display import display, Image
import time
from tqdm import tqdm
import os

# ---------- specify these args ----------
img_dir = 'train/images' # dir that saves your images
n_train = 225 # number of images
n_protein = 52 # number of proteins
# ----------------------------------------

avg_list = []
for i in tqdm(np.arange(1, n_train+1), total=n_train,
desc="Processing"):
    img_file_name = f'{i}.tiff'
    path = os.path.join(img_dir, img_file_name)
    img = imageio.v2.imread(path)
    avg = np.mean(img, axis=(1,2))
    avg_list.append([i] + avg.tolist())
avg_df = pd.DataFrame(avg_list, columns = ['id'] + ['protein' +
str(i) for i in range(1, 52+1)])
avg_df.to_csv('avg_intensity.csv', index=False)

avg_df.head()
```

Other Considerations

- Normalize the outcome data?
- Linear or non-linear regression?
- Or not regression at all?

Project Report

- A project report, 3-5 pages long, consisting of at least the following sections
  - Methods: detailed description of the method and analysis experiments
  - Results: detailed description of results
  - Discussion: caveats, future plans for improvements
- Analysis code to reproduce all results to be attached in a separate file
- A text file with predicted values for individuals in the test set

Grading of the Project

- Accuracy (15 points):
  - Very low (12)
  - Low (13)
  - Moderate (14)
  - High (15)

- Report (15 points):
  - Submitted (12)
  - Clearly/well written (+1)
  - Detailed, comprehensive, reproducible analysis (+1)
  - Novelty in method or analysis (+1)