Autoimmune Disease Machine Learning Challenge

Organized by:
Eric and Wendy Schmidt Center at the Broad Institute,
Klarman Cell Observatory at the Broad Institute,
Crunch Foundation,
Foundry Institute

October 28, 2024

Introduction

Autoimmune diseases occur when the immune system, our body's defense system, mistakenly attacks healthy cells. These diseases affect 50M people in the United States, with rates rising globally. Inflammatory bowel disease (IBD) is one of the most common types of autoimmune disease. IBD occurs when the barrier between our gut and the microbes living there breaks down, leading to the activation of the immune system in response to this barrier disruption and the invasion of microbes into the gut mucosa. The immune system is unable to resolve this tissue insult and promote tissue healing, resulting in persistent activation and chronic inflammation, with cycles of flares and remission that increase the risk of developing cancer. Before modern 20th century treatments, mortality was often greater than 50%. IBD remains a multifaceted disease that severely impacts patients' lives, with complex pathogenic pathways that make it challenging to treat.

Common symptoms of IBD include abdominal pain, diarrhea, and weight loss. A gastroenterologist can notice these clinical symptoms, but the diagnosis of IBD relies on performing an endoscopy (extracting a piece of tissue from a patient's gut) and analyzing images of the gut tissue in consultation with a highly-trained pathologist. These pathology images are essential for patient treatment as they guide not just the diagnosis of IBD, but also the choice of drugs that are best suited for the patient, and may help predict whether the patient is likely to develop colorectal cancer. Notably, the risk of colorectal cancer can be up to two-fold higher in IBD patients, but this cancer is highly treatable if detected during early screening.

Worldwide, pathologists have collected millions of gut tissue images across hospitals, making these images a treasure trove of data, and an enormous opportunity for machine learning to impact patient health.

Complementing these images collected by pathologists, the revolution in genomics over the past twenty years has enabled us to measure the activity of genes directly within these gut tissues, uncovering details a pathologist cannot identify from the images alone and providing an opportunity to unveil the pathways underlying the disease. Spatial genomics measurements will enable the next generation of IBD treatments by revealing which cells in the gut are working together to promote the disease. However, such measurements are very expensive and time-intensive to obtain.

What if we could use machine learning to connect the tissue images routinely collected by pathologists with higher-resolution but expensive, and therefore rarer, spatial genomics measurements? The resulting high-resolution and large-scale view of IBD could improve patient diagnosis, better guide the choice of drug treatment, and help identify and treat colorectal cancer earlier.

This is where you come in. We need computational approaches to connect pathology images and spatial genomics measurements. You will develop algorithms that use the images of a tissue collected by a pathologist to infer the high-resolution view of tissue visible in spatial genomics: the cells and genes driving disease. We will use your models to predict genes that are markers of potential cancerous regions in the gut, and we will then perform experiments to test these predicted genes in patient samples.

Overview of challenges

The algorithms you develop in Crunches 1 and 2 will enable researchers to gain high-resolution spatial genomics information from routine tissue pathology images. In Crunch 3, we will put your algorithms to the real test: can they discover genes, using just the routine pathology images, that identify cells important in the early steps of developing colorectal cancer?

- **Crunch 1:** Inpainting and translating held-out spatial transcriptomics data from matched pathology images.
- **Crunch 2:** Predicting never seen held-out genes in spatial transcriptomics from matched pathology images and single-cell transcriptomics data.
- **Crunch 3:** Predicting which genes mark pathologist-annotated dysplasia (i.e., pre-cancerous) regions in pathology images.

Crunch 1: Predicting spatial transcriptomics data using pathology images

Worldwide, pathologists across hospitals have collected millions of tissue pathology images from the gut to study inflammatory bowel disease (IBD), making these *H&E images* a treasure trove of data, and an enormous opportunity for machine learning to impact patient health. Hematoxylin and eosin (H&E) is the most widely used stain in medical diagnosis and has been in use since the 19th century. A pathologist uses these stains to differentiate between different parts of a cell. Each cell is surrounded by a cell membrane, which separates the interior of the cell, called the cytoplasm, from the outside environment. Inside the cell is the nucleus that contains the cell's genome, made of DNA, which holds its genetic information. Hematoxylin stains the cell nucleus a purplish blue, and eosin stains the extracellular matrix and cytoplasm pink, with other structures taking on different shades and combinations of these colors. By analyzing the overall appearance and organization of cells within a tissue, a pathologist can make a clinical diagnosis. An H&E image can be seen as a standard 3-channel RGB image. An example H&E image is provided below (Fig.1, left panel).

While H&E images guide clinical diagnosis and treatment, they do not reveal the underlying mechanisms behind disease or suggest novel treatments. Measuring gene expression in cells and in tissues is far more informative in this regard. Thanks to the ongoing revolution in genomics over the past twenty years, it is now possible to measure the expression levels of genes directly within tissues, showing us functional details a pathologist cannot see and providing an opportunity to uncover the pathways driving disease. For example, the Xenium technology (10x Genomics) can measure the expression levels of hundreds of genes directly in individual cells in their spatial tissue context. The resulting *spatial transcriptomics* data can be seen as an image with hundreds of channels, each measuring the activity of one gene in the spatial tissue context. An H&E image can be collected from this same tissue, resulting in matched datasets. An example of an H&E image (left) and the matched Xenium image for two genes (*EPCAM*, a gene that marks epithelial cells - middle; *ACTA2*, a gene that marks muscle cells - right) is provided in Fig. 1.

These novel spatial transcriptomics measurements are critical for the next generation of IBD treatments, but they are very expensive, time-intensive to obtain, and require extensive technical expertise. What if we could connect the tissue H&E images routinely collected by pathologists with these

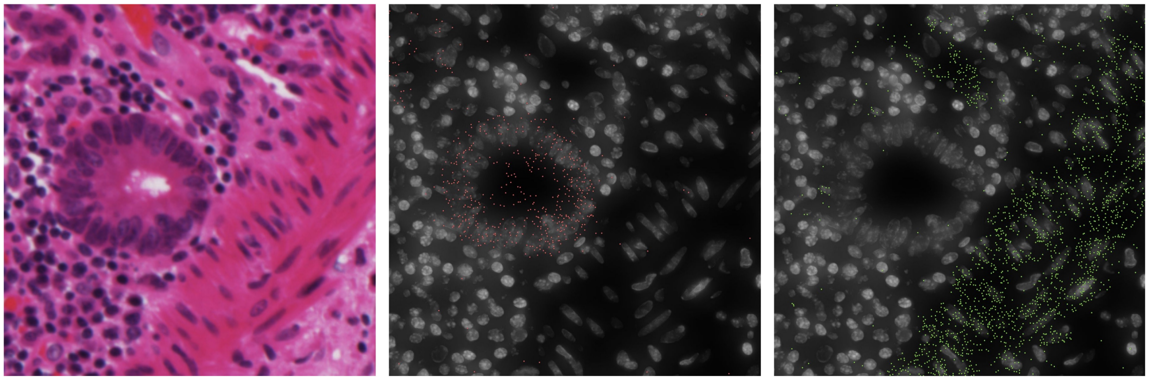

Figure 1: **H&E and Xenium spatial transcriptomics data.** Example images acquired on a small crop of mucosa and muscularis mucosae from an Ulcerative Colitis (UC) sample. On the left is an H&E stained image. The middle and right panels are spatial transcriptomic images (Xenium) showing *EPCAM* transcripts as red dots and *ACTA2* transcripts as green dots, respectively.

less-available spatial transcriptomics measurements?

In this Crunch, we will explore how well we can predict gene expression (i.e., the activity of a gene) in a tissue from a matched H&E image (Fig. 2). Specifically, we will use matched H&E images and Xenium spatial transcriptomic profiles of eight colon tissue samples, including inflamed (I) and non-inflamed (NI) tissue from human donors with ulcerative colitis (UC), the most common type of IBD, which affects the colon. We also profiled colon tissue from two patients with diverticulitis (DC), typically a milder form of colon inflammation that does not disrupt the spatial organization of the colon. This is a good reference when understanding the changes that happen during the chronic inflammation in UC, and gives your models a chance to learn the spectrum of possible colon tissue spatial organizations, from normal to diseased. The diverticulitis samples are named DC1 and DC5, and the ulcerative colitis samples are named UC1 I and UC1 NI (same patient), UC6 I and UC6 NI (same patient), UC7 I, and UC9 I.

Next, we explain an important technical aspect of how these datasets were collected. For each section of colon tissue, we generated a Xenium spatial transcriptomic dataset in which 480 genes (out of a total of around 20,000 protein-coding genes in the human genome) were profiled. After this, we prepared an H&E stain from the same tissue. Both the Xenium dataset and the H&E dataset are collected by imaging the same tissue under different microscopes. However because the colon tissue morphology may become slightly perturbed between collection of the Xenium data and preparation and acquisition of the H&E image, the images from both modalities are not perfectly aligned. Images of the same cell, collected by Xenium and by H&E, may only partially overlap.

To translate from the H&E modality (called '**HE_original**' in our dataset) to the Xenium spatial transcriptomic modality, we must first align the images from both modalities in a common coordinate framework. Here, we use an image of nuclei from a DAPI stain of the Xenium profiled tissue ('**DAPI**') to align Xenium to an image of nuclei from the H&E. Note that Xenium itself is not a traditional image and it is easier to align these two modalities through the auxililary DAPI image, as outlined in the next paragraph with further background. To anchor and align these two modalities, we provide the nuclear segmentation masks for the H&E (using the hematoxylin stain, '**HE_nuc_original**') and Xenium modalities (using the DAPI stain, '**DAPI_nuc**'). These masks outline the boundaries of the individual nuclei. **Importantly, we have already aligned the modalities for you, using nuclear segmentation masks for the H&E and Xenium data as the anchor with which modalities are translated (see section on Dataset for details)**. This alignment maps H&E images from the original spatial coordinates (**∗_original**) to registered coordinates (**∗_registered** that match those in the DAPI images. Accordingly, you can use the registered H&E image ('**HE_registered**') and the registered H&E segmentation mask ('**HE_nuc_registered**') to map between the H&E and Xenium data (represented by **DAPI**).

Here, we provide additional details on this segmentation: For the H&E, hematoxylin specifically stains the nuclei blue, and in the Xenium data, an applied DAPI stain makes the nuclei appear blue under fluorescent excitation. By aligning the nuclei of cells in the Xenium data and the nuclei of the same cells in the H&E data, the two modalities become aligned and the tissue coordinates in the H&E image map accurately to the gene transcripts measured by Xenium in the exact same tissue area. No alignment is perfect but in general, the images are well aligned. Furthermore, all validation and test set regions have high-quality alignments between Xenium and H&E data. **It is your decision whether you use our provided alignments for training your models or work on improving the alignment of the two nucleus segmentation masks.** Note that while nuclei can be identified in the H&E and Xenium images, cellular segmentation masks, which delineate the boundaries of the individual cells, are not readily available from this data and would need to be learned building off the nucleus segmentation masks. This is another path you could consider to

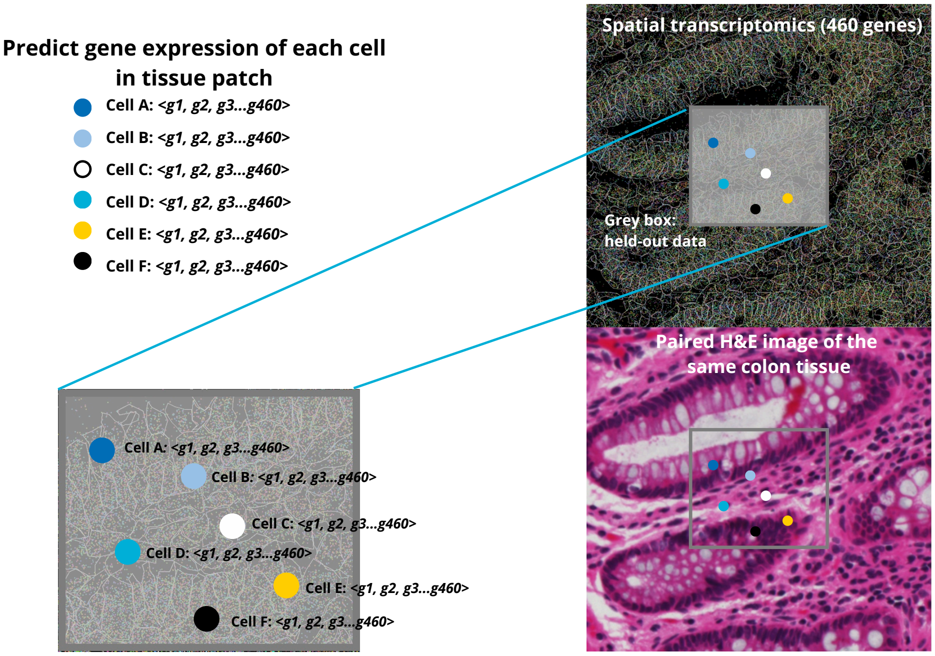
Predict gene expression of each cell in tissue patch
Cell A: <g1, g2, g3...g460>
Cell B: <g1, g2, g3...g460>
Cell C: <g1, g2, g3...g460>
Cell D: <g1, g2, g3...g460>
Cell E: <g1, g2, g3...g460>
Cell F: <g1, g2, g3...g460>
Spatial transcriptomics (460 genes)
Grey box: held-out data
Paired H&E image of the same colon tissue
Cell A: <g1, g2, g3...g460>
Cell B: <g1, g2, g3...g460>
Cell C: <g1, g2, g3...g460>
Cell D: <g1, g2, g3...g460>
Cell E: <g1, g2, g3...g460>
Cell F: <g1, g2, g3...g460>

Figure 2: **Predicting spatial transcriptomics data from an H&E image (Crunch 1).** For each tissue, we provide spatial transcriptomics data (right), which includes the gene expression of 460 genes in each cell and a segmentation mask, paired with an H&E image. Within each tissue, we hold out patches of spatial transcriptomics data (grey box) and ask you to predict the expression of those 460 genes in held out cells (lower left) using the paired H&E image and the spatial transcriptomic data from the surrounding tissue.

better leverage this data, as you design and train your models. However, evaluation will be performed based on predictions for the nuclear-segmented regions only. The object `cell_id-group` described in Dataset below, as well as all files containing segmentation masks, are filtered to contain exactly the same cells.

In this crunch, we will hold out tissue patches of Xenium spatial transcriptomic data with different sizes, and you will predict the gene expression profile for each nucleus in these held-out patches using H&E images of the whole tissue slide and Xenium data of the surrounding tissue (effectively holding out patches of spatial transcriptomics information). The larger tissue patches will sample all colon tissue layers, while the smaller tissue patches will sample tissue-specific cellular structures (Fig.3). This inpainting task will help us predict gene expression from expensive spatial transcriptomics experiments using much cheaper H&E images.

Crunch 1 presents an **in-distribution** prediction task with respect to the 460 genes studied (Fig. 2). In Crunch 2 we will consider the **out-of-distribution** task of predicting the expression of genes that are not provided as measurements in the Xenium spatial transcriptomic training data (Fig.4).

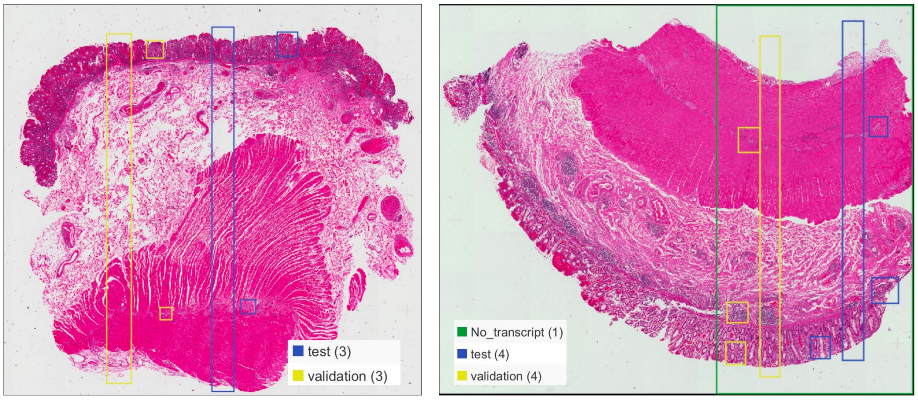

Figure 3: **Spatial transcriptomics of held-out tissue patch regions**. Examples of spatial transcriptomics held out regions, on the left, for the UC1 I sample, and on the right for the UC7 I sample (UC: ulcerative colitis; I: inflamed). We have designed the validation (yellow boxes) and test (blue boxes) regions to evaluate your model's performance on both global and local spatial prediction problems. For the UC7 I sample the held-out tissue patches are not immediately adjacent to tissue regions with measured Xenium spatial transcriptomic data, with the green box indicating this much larger tissue area with no available transcriptomics data, named No_transcript-train.

Dataset:

For each colon tissue section, we provide you with the Xenium spatial transcriptomic data, paired H&E image, nucleus segmentation masks, DAPI image, and registered H&E images as a `SpatialData` object stored in a `zarr` file. You can read more about the SpatialData object here, and the structure of a `SpatialData` object is shown below. Note that all references to spatial coordinates in the following files are in a common coordinate system, except for the original H&E images, which are measured in their own pixel coordinate system. The image registration process is described after the data format specifications. In addition, we provide you with a jupyter notebook through the Crunch website that includes code for loading these objects, interacting with the data, visualizing the transcriptomic data and H&E image, and the scoring function used for evaluation of your model's predictions.

```
SpatialData object structure
 Images
      'DAPI': DAPI image (validation and test tissue patches are removed)
      'DAPI_nuc': DAPI nucleus segmentation
      'HE_nuc_original': H&E nucleus segmentation on original image
      'HE_nuc_registered': H&E nucleus segmentation on registered image (registered to DAPI image)
      'HE_original': H&E original image
      'HE_registered': H&E registered image
      'group': Defining train(0)/validation(1)/test(2), No_transcript-train(4) tissue patches
      'group_HEspace': Defining train(0)/validation(1)/test(2), No_transcript-train(4)
tissue patches on the H&E image
 Points
      'transcripts': DataFrame for each transcript (containing x,y,tissue patch,z_location,
    feature_name,transcript_id,qv,cell_id columns)
 Tables
```

```
      ‘anucleus’:  AnnData contains .X, .layers[‘counts’], .obsm[‘spatial’]
      ‘cell_id-group’: AnnData only contains .obs DataFrame for mapping of cell_id
       to region.
with coordinate systems:
    ‘global’, with elements:
       DAPI (Images), DAPI_nuc (Images), HE_nuc_original (Images), HE_nuc_registered
           (Images), HE_original (Images), HE_registered (Images), group (Images),
           group_HEspace (Images), transcripts (Points)
    ‘scale_um_to_px’, with elements:
       transcripts (Points)
```

1. Original, full-sized H&E image (1̃0mm × 22mm) with 3 channels provided under the key **HE_original**. These images are *not* registered to the coordinate system in the Xenium spatial transcriptomic data. Also note that the original H&E image actually consists of two separate tissue sections that were placed on the same slide and profiled together by Xenium. If you decide to work with the original H&E image, be sure to choose the correct tissue section.

2. Spatial transcriptomics data of 460 genes (channels); 20 genes are held-out and used for out-of-distribution predictions in Crunch 2. In accordance with the file structure of **transcripts** from

   `https://cf.10xgenomics.com/supp/xenium/xenium_documentation.html`,

   this transcript-level data is supplied as a table under the key **transcripts** indexed by individual transcripts with the following columns:

   (a) transcript_id: unique ID of transcript

   (b) cell_id: unique ID of cell/nucleus, as also referenced in **cell_id-group**. Transcripts that do not lie within a nuclear segment are labeled as ”0”.

   (c) feature_name: gene name (out of the 460 channels).

   (d) x: X location (unit: μm, pixel size: 0.2125 $\mu m/px$). You need to convert to px unit to find exact pixels in the Xenium coordinate system, e.g the **DAPI** and **DAPI_nuc** image, etc..

   (e) y: Y location (unit: μm).

   (f) z_location: Z location (unit: μm).

   (g) qv: Phred-scaled quality value estimating the probability of incorrect call for each transcript.

   (h) Tissue patch region: Always 0. You are only provided transcripts from the regions meant to be used in training your models.

   Transcripts that correspond to held-out validation (1) and test (2) regions are not included in this file, and you will provide them as predictions.

3. Segmentation masks of nuclei from H&E images; non-zero integers represent pixels inside of the nuclei where each nucleus segment corresponds to one integer (numbering starts at 1), while 0s represent pixels outside of the nuclei, provided with key **HE_nuc_original**.

4. We provide a simpler form of the **transcripts** data introduced above under the **anucleus** key. For each nucleus, transcripts for each gene (measured by spatial transcriptomics) are summed and provided as a table in the anndata format. Read about this format and the `scanpy` API here. In this table (`anucleus.X`), every observation (cell_id) is a segmented nucleus and the features are the summed gene expression of each of the 460 genes detected in the

nucleus. Only cell_ids belonging to the training set are stored, as your task is to predict the gene expression in the validation and test sets. The spatial coordinates of the center of the nucleus are provided in (`anucleus.obsm["spatial"]`, namely the x and y coordinates based on the registered images (**DAPI**), and the cell_id is provided in (`anucleus.obs["cell_id"]`). The gene expression data (`anucleus.X`) is log1p-normalized, which means that the original gene expression counts per nucleus are divided by the total counts in the nucleus, multiplied by 100, and then log1p transformed. Specifically, the `scanpy` code for doing the normalization is: sc.pp.normalize_total(anucleus, inplace=True, target_sum=100) and sc.pp.log1p(adata). The raw aggregated gene counts are stored in a separate slot in the object and can be accessed under `anucleus.layers["counts"]`. Held-out nuclei are listed under the key **cell_id-group** obs dataframe with group validation(1)/test(2). Note that `anucleus.X` could be computed from the raw spatial transcriptomics data, `adata.layers['counts']`, and the segmentation masks with key **DAPI_nuc**. We provide the gene-expression table `anucleus.X` for simplicity.

5. DAPI image (1 channel), in Xenium coordinate system, provided under the key **DAPI**.

6. Segmentation mask of nuclei from DAPI image, provided under the key **DAPI_nuc**.

7. Registered H&E image (3 channels), in Xenium coordinate system, provided under the key **HE_registered**.

8. Segmentation mask of nuclei from registered H&E image, provided under the key **HE_nuc_registered**.

9. Each pixel in **group** is assigned an integer value representing train(0) / validation(1) / test(2) / No_transcript-train(4), based on the registered coordinate system. No_transcript-train(4) covers a much larger area of the H&E image, where no transcripts are provided in training, but within this area there are validation and test regions.

10. Each pixel in **group_HEspace** is assigned an integer value representing train(0) / validation(1) / test(2) / No_transcript-train(4), based on the original H&E coordinate system.

11. **cell_id-group** contains a table showing the mapping of **cell_id** to a string representing train / validation / test / No_transcript-train. Cell IDs start at 1.

Description of the image registration process: We performed two steps of registration to match the original H&E image to the Xenium coordinate system (i.e. DAPI image). In the first step, we found matched nuclei as landmarks in both H&E and DAPI images, and an affine transformation was used to transform the H&E image. After this, we used nucleus segmentation from registered H&E and DAPI images to find local shifts at the 1024px*1024px patch level. A displacement field was generated using all the local shifts to transform the H&E image further. Applying this two-step strategy produced the final registered H&E image and matched cell_id in all nucleus segmentations provided. Remember, we provide this registration for your convenience, but it is not perfect and you have the option to modify it if you think this will improve training of your model.

Participant output:

For each of the eight tissue samples, provide gene expression predictions for each held-out nucleus (`"group == 'test' or group == 'validation'"`) as a table **rounded to 2 decimal points** as a `csv` file with nucleus IDs as row names and 460 gene features as column names. Make sure your predictions for each nucleus are log1p-normalized as in `anucleus.X`. Also, make sure your file can be read in using the pandas command `pd.read_csv(FILENAME, header=0, index_col=0)`. We also provide example output files for each tissue sample in **validation-test-example-crunch1.zip**.

Evaluation:

You will have multiple opportunities to evaluate your model's predictive performance on a validation dataset, before submission of your test dataset predictions. There will be two validation checkpoints, occurring on November 30th (Eastern Time 17:59) and December 30th (Eastern Time 17:59), before you submit your final test dataset predictions on January 31st (Eastern Time 17:59). output all the held-out data for the checkpoints(both test and validation groups, as indicated above) and we'll subset the outputs to evaluate your model on the validation group. We have designed the validation and test datasets to evaluate your model's performance on both global and local spatial prediction problems. The global hold-out is a rectangular "core" tissue patch, extending from the innermost layer of the colon (mucosa) to its outermost layers (muscle and serosa). This tests your model's ability to recognize the overall spatial organization of the colon and how it changes from normal to inflamed disease. The local hold-outs are much smaller tissue patches that represent specific cellular organizations and interactions within the colon layers including: the colon mucosal layer, lymphoid aggregates (large groups of immune cells), and the myenteric plexus (neurons controlling colon muscle movement). Each of the eight tissue sections will have roughly the same number of global (1) and local (2) tissue patches across both the validation and test datasets, as shown in Fig. 3 left.

We want to highlight validation and test datasets that we expect to be the most challenging for your predictions, and are also very important if you are (hopefully) planning to complete Crunch 3. For tissue section (UC7 I), the held-out tissue patches are not immediately adjacent to tissue regions with measured Xenium spatial transcriptomic data (Fig. 3 right). For tissue section (DC1), no spatial transcriptomic data is provided. While the other inpainting predictions you make can be considered interpolation, these are more difficult tests of how well your model can extrapolate to tissue regions where Xenium transcriptomic data is not spatially adjacent. Furthermore, you will encounter a similar situation in Crunch 3, where we will provide you the H&E image and Xenium data for one half of the tissue section with noncancerous mucosa, and ask you to make predictions in the other half of the tissue section, where only H&E image data is available. These tissue areas are labeled **No_transcript**-train(4), as no transcripts are provided here, however the H&E images can still be used in training your models.

Your predictions $\hat{X}$ will be evaluated based on the mean squared error to the log1p-normalized validation/test data X:

$$L = \frac{1}{N_{\text{nuclei}}} \sum_{i \in \text{nuclei}} \frac{1}{N_{\text{genes}}} \sum_{j \in \text{genes}} (\hat{X_{ij}} - X_{ij})^2$$

This score reflects the output of the scoring function in the example notebook. Note that we will compute this score separately (i) for each of the global and local hold out tissue patch regions (ii) within each of the eight tissue sections. We will take a mean score over all nuclei in the cells of each given tissue patch region. Tissue sections do not necessarily contain equal representation of all tissue patch regions, thus necessitating a weighting step to avoid over-representation of commonly occurring cell types and regions in the final score. We will then compute a weighted aggregate of these separate global and local scores that corrects for sizes of different tissue regions. You do not have access to this weighting but you will be able to validate model fits at the checkpoints. Finally, we will take the mean of the scores in the global and the local tasks to generate the final score.

External resources:

The application of external resources (e.g., external gene expression datasets including the dataset provided in Crunch 2, external H&E images or pretrained embeddings, etc.) is allowed; however, all external resources must be published or in the public domain and properly credited. In addition, you can optionally use the Foundry computing environment, which provides $10 USD of GPU time and a python environment. Depending on the Foundry market, we estimate this may be about 10 hours of runtime on an instance with an A5000 GPU.

Crunch 2: Predicting unmeasured genes in spatial transcriptomics from matched pathology images and single-cell transcriptomics data

It is very challenging and expensive to spatially measure the gene expression of all protein-coding genes in the genome with high resolution in intact tissues. However, we can measure the expression of all genes in single cells dissociated from intact tissues cells using single-cell transcriptomics technologies. In the spatial transcriptomics data that we consider here, 480 genes (out of all possible protein-coding genes) were selected based on prior biological knowledge that they mark different cell types, are genetically associated with IBD, or are involved in signaling between cells. However, for many cell states of interest, such as early cancer states, informative marker genes are unknown and need to be identified, which will be the subject of Crunch 3.

In this Crunch, you will predict the spatial expression of 2,000 protein-coding genes that are not in the Xenium spatial transcriptomic training data (Fig. 4). To enable this prediction, we provide separate single-cell transcriptomics (scRNA-Seq) data that is matched to Xenium data. The scRNA-Seq data comprise the expression of 18,615 protein-coding genes including the 460 genes from Crunch 1 and the 2,000 genes that need to be predicted. The set of 2,000 genes also includes the 20 genes previously measured by Xenium but held-out from the Spatial Data object, which will be used for evaluation. Because scRNA-Seq measures many more genes than a Xenium measurement, the resulting transcriptional profiles for every single cell are more informative, and by clustering cells based on these profiles, we can identify all cell types that make up the colon tissue. This approach has been used extensively over the past decade to comprehensively identify and describe cell types found in human tissues. As an example, visit the Human Cell Atlas, where tissues from all human organs, including the colon, have been profiled by scRNA-Seq.

Both Xenium and scRNA-Seq were used to measure similar pieces of colon tissue, so with both modalities, we are detecting the same cell types with similar gene expression profiles. Because the same cell types are measured by both technologies, the much larger set of genes measured by scRNA-Seq can help fill in the genes not measured by Xenium. Although scRNA-Seq detects many more genes in each cell, the spatial resolution is lost because scRNA-Seq requires dissociating the tissue into single cells and encapsulating each cell into a droplet in order to perform high-throughput RNA sequencing. Still, one can leverage the scRNA-Seq data to learn how the unmeasured genes co-vary with the 460 genes that are measured in the Xenium spatial transcriptomics training data. We will evaluate your predictions based on Spearman's rank correlation coefficient. There will not be any opportunities for validation in Crunch 2.

Dataset:

Note that all references to spatial coordinates in the following files are in the same coordinate system. For all but the original H&E images, they are measured in the Xenium coordinate system, the H&E images are registered with this coordinate system and presented here as transformed coordinates.

1. Xenium SpatialData object in **.zarr** format, as described in Crunch 1.
2. Single-cell RNA-seq (scRNA-Seq) data of colon tissue samples similar to the samples profiled by Xenium spatial transcriptomics, with the expression of the protein-coding genes that need to be predicted (N=18,615), which include the 460 genes in the Xenium data. We have collated

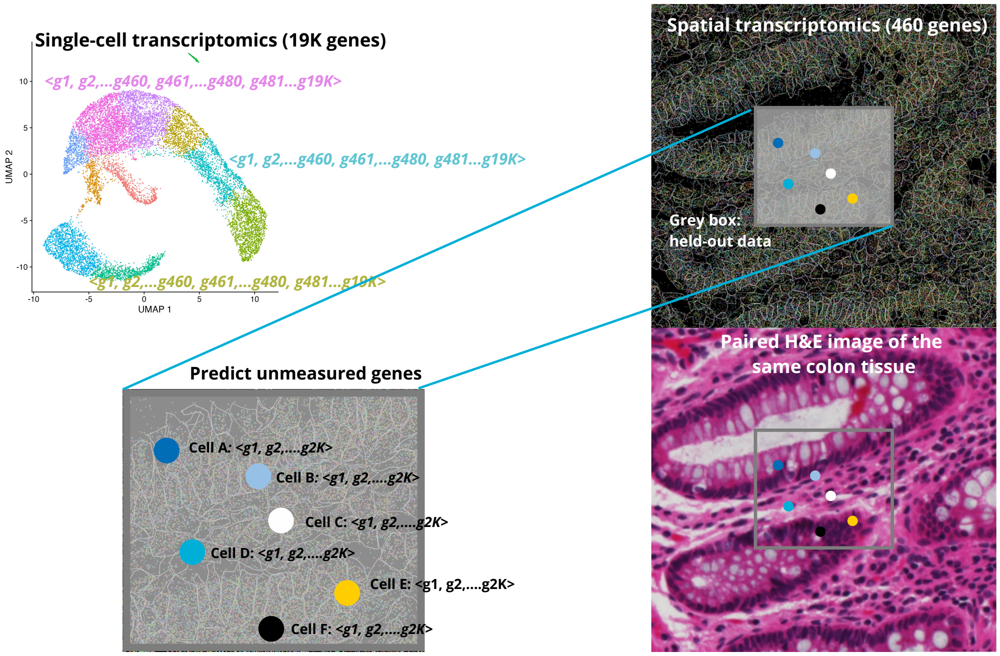

Figure 4: **Predicting unmeasured genes in spatial transcriptomics from matched pathology images and single-cell transcriptomics data (Crunch 2).** Within each tissue patch region, predict the gene expression of 2,000 protein-coding genes (lower left) using the spatial transcriptomic training data (expression of 460 genes per cell) (right), segmentation mask, paired H&E image, and single-cell RNA-seq data on 18,615 genes (upper left) obtained from similar colon tissues. Prediction accuracy will be evaluated based on the expression of 20 genes that were measured, but held out, from the spatial transcriptomics training data ($g461$,....$g480$ in the upper left image of the single-cell transcriptomic data). This figure is also shown in the accompanying crash-course video lectures, but the gene numbers are updated here to the final numbers used in this challenge.

four scRNA-Seq datasets that cover the cell types and states found in healthy and diseased colon tissue including: an extensive atlas of ulcerative colitis patients including inflamed, non-inflamed, and healthy colon tissue defined based on pathologic evaluation (UC), an atlas of the enteric nervous system including the glial cells and neurons innervating the colon (ENS), and an atlas of the colon muscle layer (muscle). How you integrate the scRNA-Seq datasets with the Xenium data is your decision. However, we expect the cell type annotation information may be helpful as it is a grouping of cells by similar gene expression profiles indicating how genes covary with one another in a cell type. The scRNA-Seq datasets are provided as an `anndata` object stored in an `h5ad` file **Crunch2_scRNAseq.h5ad**, which includes cell meta data in an `obs` dataframe indicating for each cell, its cell type (`adata.obs["annotation"]`), in which study the cell was profiled (`adata.obs["study"]`), from which individual the cell was isolated (`adata.obs["individual"]`), and the disease status of the individual (`adata.obs["status"]`). This data (`adata.X`) is log1p-normalized, which means that the original gene expression counts per cell is divided by the sum of counts per cell, multiplied by 10,000 and then log1p transformed. The raw gene counts are stored in a separate slot in the `anndata` object and can be accessed under `adata.layers["counts"]`.

Participant output:

For each of the eight tissue samples, provide gene expression predictions **rounded to 2 decimal points** for each held-out nucleus as a DataFrame with nucleus IDs from the tissue as row names and 2,000 gene features as column names. The DataFrame dimension is $n_{\text{nuclei}} \times 2,000$, where n_{nuclei} is the number of nuclei included in the corresponding validation or test set and 2,000 is the number of protein-coding genes, including the 20 held-out genes, for which to predict expression. Make sure your predictions are log1p-normalized as in `adata.X` from **Crunch2_scRNAseq.h5ad** and the predicted values are limited to **2** decimal points.

Evaluation:

In this Crunch you will not have checkpoints, as in Crunch 1, to assess your performance on the validation set. You will submit all predictions for validation and test set regions by January 31st (Eastern Time 17:59). Your predictions will be subset to 20 genes, which we held-out from the Xenium spatial transcriptomic dataset. Your predictions $\hat{X}$ will be evaluated on these 20 genes based on (i) the mean of Spearman's correlation to the log1p-normalized test data X for test cells with any non-zero gene expression in the 20 target genes, and (ii) and an auxiliary score for test cells that were all-zero gene expression on those genes, for which Spearman's correlation cannot be meaningfully evaluated:

Score for cells with non-zero gene expression entries will be:

$$L_1 = \sum_{i \in \text{nuclei non-zero}} r_s(\hat{X}_i, X_i) \tag{1}$$

Cells with all-zero gene expression in these 20 target genes will be evaluated based on an auxiliary score or removed from scoring.

As in Crunch 1, the validation and test sets consist of global and local tissue patch regions, which are identical to the tissue patches tested in Crunch 1. In these patches, you receive the tissue H&E image, which is surrounded by both the remaining H&E image and the Xenium spatial transcriptomics measurements for 460 genes.

External resources:

The application of external resources (e.g., external gene expression datasets including the dataset provided in Crunch 2, external H&E images or pretrained embeddings, etc.) is allowed; however, all external resources must be published or in the public domain and properly credited. In addition, you can optionally use the Foundry computing environment, which provides $10 USD of GPU time and a python environment. Depending on the Foundry market, we estimate this may be about 10 hours of runtime on an instance with an A5000 GPU.

Crunch 3: Identifying gene markers of pre-cancerous tissue regions in IBD

The risk of colorectal cancer can be up to two-fold higher in IBD patients, but this cancer is highly treatable if detected during early screening. Based on H&E images, pathologists can identify and label regions of dysplasia in the colon tissue, which are regions containing abnormally appearing cells that may develop into cancer. Dysplasia originally arises in the normal, epithelial cell lining of the colon. However, the sets of genes expressed in cells from dysplasia tissue regions are unknown. If we can identify the genes, or gene expression programs, driving dysplasia, we can have a better understanding of the functional details and the molecular pathways underlying this early cancer process, thereby improving the diagnosis and treatment of colorectal cancer.

In Crunch 3, the goal is to design a panel of genes that best distinguishes dysplasia regions from noncancerous mucosa regions (Fig. 5). We will provide you with H&E images that have been labeled by a pathologist to indicate dysplasia regions and noncancerous mucosa regions. You will rank all protein-coding genes by how well you expect them to discriminate between dysplasia and noncancerous tissue regions. If you have participated in Crunch 2, you may choose to use your trained model to make gene expression predictions on these regions and then design a gene panel based on these predictions. Also for those who participated in Crunch 1 or 2, it is important to note that the setting in Crunch 3 is similar to the extrapolation test set predictions you had previously made. Here, we provide you Xenium spatial transcriptomic data and the corresponding H&E image for one half of the tissue section, but for the half of the tissue section where dysplasia has been annotated, we only provide you the H&E image. For those who did not participate in Crunch 2, you can design a gene panel from scratch using biological understanding or other approaches. Regardless of your chosen approach, you are required to provide a justification for how you constructed your gene panel. You are also required to participate in peer-reviewing three submissions of other participants based on their justifications of their gene panel design.

We will select a subset of genes from participants' output as our new gene panel and perform a new spatial transcriptomics experiment (see "Validation experiments" below). We will evaluate how these selected genes discriminate between cells in noncancerous mucosa and dysplasia regions, rewarding candidates for identifying distinct gene programs (see "Evaluation' below).

Dataset:

1. We provide H&E images from two colon tissue sections from the patient with dysplasia: the first H&E image is collected post-Xenium and only includes the noncancerous mucosa (already provided in Crunch 1), and the second H&E image is the entire colon tissue section including both the dysplasia region and noncancerous mucosa (UC9_Infl-crunch3-HE.tif). For each of the two tissue sections, we provide the H&E image, the nucleus segmentation mask, and the defined tissue region mask as three tiff files. Regions of dysplasia and regions of noncancerous mucosa tissue are marked in the tissue region mask (UC9_Infl-crunch3-HE-dysplasia-ROI.tif). These annotations are stored as a channel with categorical values: 1 indicates noncancerous mucosa, 2 indicates dysplasia, and 0 indicates other tissue regions.

2. We provide single-cell RNA-seq (scRNA-Seq) data of colon tissue samples with and without dysplasia. This data provides the gene expression of 18,615 protein-coding genes, including the 460 genes measured in the Xenium data. While we do not have scRNA-Seq data collected from the individual with diagnosed dysplasia (UC9 I), a separate study recently reported single-cell transcriptomic profiling of healthy colon, polyps, and colorectal cancer. Polyps are

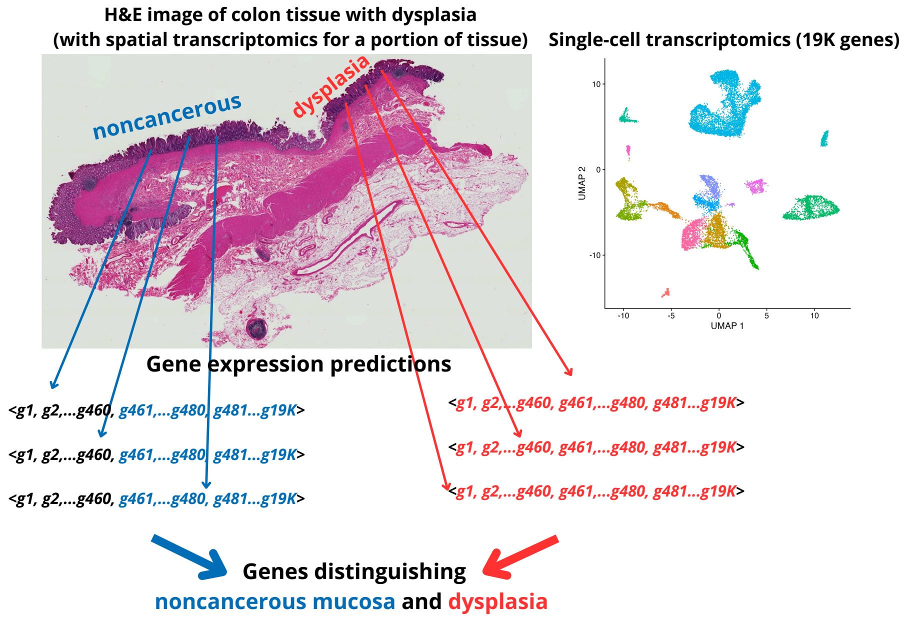
H&E image of colon tissue with dysplasia
(with spatial transcriptomics for a portion of tissue)
Single-cell transcriptomics (19K genes)
noncancerous
dysplasia
10
0
-10
UMAP 2
-10
-5
0
5
10
UMAP 1
Gene expression predictions
<g1, g2,...g460, g461,...g480, g481...g19K>
<g1, g2,...g460, g461,...g480, g481...g19K>
<g1, g2,...g460, g461,...g480, g481...g19K>
<g1, g2,...g460, g461,...g480, g481...g19K>
<g1, g2,...g460, g461,...g480, g481...g19K>
<g1, g2,...g460, g461,...g480, g481...g19K>
Genes distinguishing
noncancerous mucosa and dysplasia

Figure 5: **Identifying gene markers of pre-cancerous tisue regions in IBD (Crunch 3).** Given scRNA-Seq data from colon tissue samples with dysplasia (upper right), Xenium spatial transcriptomic data for the noncancerous half of a tissue section paired with an H&E image, and an H&E image of the entire tissue section with both noncancerous and dysplasia regions (upper left), predict the gene expression for cells in both halves of the tissue section, where noncancerous and dysplasia regions have been annotated. Rank the 18,615 protein-coding genes by how well you expect them to discriminate between dysplasia and noncancerous tissue regions, and provide a justification for how you constructed your ranking. You will be asked to evaluate three submissions of other participants based on their justifications of their gene panel design.

abnormal growths of the epithelial cells lining the colon, are typically non-cancerous, and often present dysplasia under the microscope. These polyps exist on a continuum of transcriptional states, from normal colon to colorectal cancer, and will inform you on the gene expression programs active during dysplasia. Importantly, we note that a baseline classification solution to this Crunch would be to work directly with this provided scRNA-Seq data and to compare gene expression programs between dysplasia and noncancerous mucosa tissue regions, as was performed in the initial study. We expect that your models developed in Crunches 1 and 2, using Xenium spatial transcriptomic data and H&E images, can outperform this baseline.

The scRNA-Seq dataset is provided as an `anndata` object stored in a `h5ad` file **Crunch3_scRNAseq.h5ad** that includes cell meta data in an `obs` dataframe indicating for each cell, its cell type (`adata.obs["annotation"]`), from which individual the cell was isolated (`adata.obs["individual"]`), the disease status of the colon tissue specimen (`adata.obs["status"]`), and whether a pathologist diagnosed dysplasia in the tissue (`adata.obs["dysplasia"]`). The disease status can be: Normal, Unaffected tissue, Polyp, and Adenocarcinoma (cancer). The dysplasia status can be: y (yes, dysplasia detected), n (no, dysplasia not detected), or ND (not provided in the study).

This data (`adata.X`) is log1p-normalized, which means that the original gene expression counts per cell is divided by the sum of counts per cell, multiplied by 10,000 and then log1p transformed. The raw gene counts are stored in a separate slot in the `anndata` object and can be accessed under `adata.layers["counts"]`.

3. Output features (genes names) in the context of example output file **example_output-crunch3.csv** where the row names are the 18,615 genes which are to be ranked by how well each gene is predicted to distinguish regions of dysplasia from noncancerous mucosa.

Participant output:

1. Prediction of rank for each gene as a table of size $18,615 \times 1$ with gene IDs as row names and the predicted rank of the gene from 1 (best at distinguishing dysplasia from noncancerous mucosa) to 18,615 (worst) as entries. See **example_output_crunch3.csv** for more details. The genes must be ordered as in **example_output_crunch3.csv** and ties are not allowed.
2. 1 page report detailing the justification for your gene panel design, in the following format:
   - at least 1 paragraph (5-10 sentences) describing how your method works
   - at least 1 paragraph (5-10 sentences) describing the rationale behind the method for your gene panel design
   - 1 paragraph (5-10 sentences) describing the datasets and any other resources used

   A list of references can be included and does not count towards the 1-page limit.

Mandatory participation in peer reviewing

In order to qualify for prizes in Crunch 3, you are required to review three submissions of gene panel designs from other participants based on the justification they have provided. In particular, you will be assigned three submissions and are expected to rank them on a 1-3 scale (1-excellent justification, 2-adequate justification, 3-poor justification) and provide a short explanation of your ranking of **200-400 words** covering the following aspects:

- rationale of design

- novelty of design
- whether the submitted justification complies with the required format.

Experimental validation of dysplasia gene panels

We will select 500 genes for experimental evaluation, as follows. There are two routes to have your genes selected.

- **Route 1:** The top 5 performing teams in Crunch 2 who also participate in Crunch 3 will have up to 50 of their top genes in their ranked lists included in the panel, resulting in a total of up to 250 genes (there will likely be some overlap across teams). The rationale here is that models that are good at predicting gene expression from H&E should be useful for selecting genes that correlate with new, unseen tissue structures.
- **Route 2:** The top 5 performing teams in Crunch 3, based on a combination of peer and expert committee reviews, will have up to 50 of their top genes included in the panel, resulting in up to 250 additional genes.

We will order a Xenium gene panel with at most 500 genes from the two participation routes above. We will reserve a small number of genes in the panel to specifically identify important cell types in the colon. We will use this Xenium gene panel to carry out spatial transcriptomics measurements of a previously unmeasured colon tissue section, with pathologist diagnosed dysplasia.

Evaluation:

For each team, we will choose the top 50 highest-ranked genes from their ranked list as described above. For the top 5 teams (either by Route 1 or Route 2 above), these 50 genes will be the top 50 genes in their ranking; for other teams, these top 50 genes will likely be ranked differently. You will submit your ranked gene panels by January 31st (Eastern Time 17:59).

We will compute a ranking of all submissions as follows. Using the noncancerous mucosa and dysplasia regions we annotated, we will train k-fold cross-validated logistic regression classifiers to distinguish noncancerous and dysplasia regions using the 50 genes selected from each team as features. We will rank participants by classification accuracy (higher is better). For each nucleus i, denote the true classification label as y_i and the predicted label as $\hat{y_i}$. Then the accuracy for each test set t is defined as follows:

$$L_{\text{test set t}} = \frac{1}{N^t_{\text{nuclei}}} \sum_{i \in \text{nuclei}} \blacksquare(\hat{y_i} = y_i),$$

where $\blacksquare$ denotes the indicator function. In each cross-validation run, we will leave out one fold as test set and train a classifier on the remainder of the data. We will then calculate the accuracy on the left out test set. We will repeat this for all k folds and use the average accuracy over all runs to rank the teams. See below for a pseudo-code for the cross-validation strategy.

We will also compute a diversity ranking to encourage the inclusion of genes associated with different biological functions. Each of the 500 genes will be normalized by z-score. The principal components (PCs) of the normalized data with all 500 genes will be computed. For each team, we will then compute the projection of the 50-gene subset to the PCs. The sum of the PC scores is the diversity score. We will compute an overall ranking by weighting the cell classification and diversity rankings. The ranking will be mainly determined by the classification accuracy as described above and supplemented by diversity rankings.

Algorithm 1 k-fold cross-validation

```
Divide the data into $k$ folds, stratified by observed labels
for $t$ fold in $k$ folds do
    Train logistic classifier on all data except for $t$ fold
    Calculate accuracy on $t$ fold: $L_{\text{test set t}}$
end for
Calculate overall accuracy by averaging over all $k$ folds
```

External resources:

The application of external resources (e.g., external gene expression datasets including the dataset provided in Crunch 2, external H&E images or pretrained embeddings, etc.) is allowed; however, all external resources must be published or in the public domain and properly credited. In addition, you can optionally use the Foundry computing environment, which provides $10 USD of GPU time and a python environment. Depending on the Foundry market, we estimate this may be about 10 hours of runtime on an instance with an A5000 GPU.

References

Below are a few references meant to provide more background and some of the approaches researchers are applying in the fields relevant to these Crunches. This is not meant to be an exhaustive list and many important works are not listed here. We may add more informative references in response to your questions over the next three months.

Single cell RNA-Seq colon tissue datasets

- Intra- and Inter-cellular Rewiring of the Human Colon during Ulcerative Colitis
- The Human and Mouse Enteric Nervous System at Single-Cell Resolution
- Single Nucleus Sequencing of Human Colon Myenteric Plexus-Associated Visceral Smooth Muscle Cells, Platelet Derived Growth Factor Receptor Alpha Cells, and Interstitial Cells of Cajal
- Single-cell analyses define a continuum of cell state and composition changes in the malignant transformation of polyps to colorectal cancer

Spatial transcriptomic colon tissue datasets

- Organization of the human intestine at single-cell resolution
- Single-cell and spatial multi-omics highlight effects of anti-integrin therapy across cellular compartments in ulcerative colitis

H&E histopathology tissue foundation models

- A foundation model for clinical-grade computational pathology and rare cancers detection
- A whole-slide foundation model for digital pathology from real-world data
- A pathology foundation model for cancer diagnosis and prognosis prediction
- Towards a general-purpose foundation model for computational pathology
- A multimodal generative AI copilot for human pathology
- Hibou: A Family of Foundational Vision Transformers for Pathology
- H-optimus-0
- HEST-1k: A Dataset for Spatial Transcriptomics and Histology Image Analysis
- Virchow2: Scaling Self-Supervised Mixed Magnification Models in Pathology

Objects and APIs for spatial and single cell analysis

- SpatialData - An open and universal framework for processing spatial omics data.
- Squidpy - Spatial Single Cell Analysis in Python.
- Scanpy - Single-Cell Analysis in Python.

Optional Review Articles

This challenge draws on many different subject areas, which are covered in the three introductory crash course lectures. To supplement this, we provide scientific review articles on these subject areas, which can give you a more detailed perspective and point you to other relevant datasets and data modalities. **Reading these articles is not necessary to complete the challenge, but we believe these can be a helpful resource**.

- Pathway paradigms revealed from the genetics of inflammatory bowel disease
- The expanding vistas of spatial transcriptomics
- Exploring tissue architecture using spatial transcriptomics
- Current best practices in single-cell RNA-seq analysis: a tutorial
- Single-cell transcriptomics to explore the immune system in health and disease