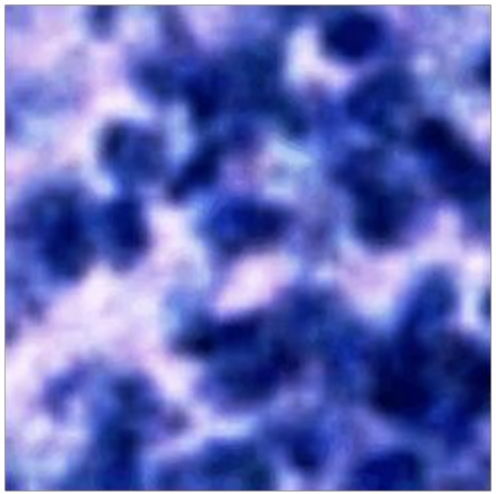
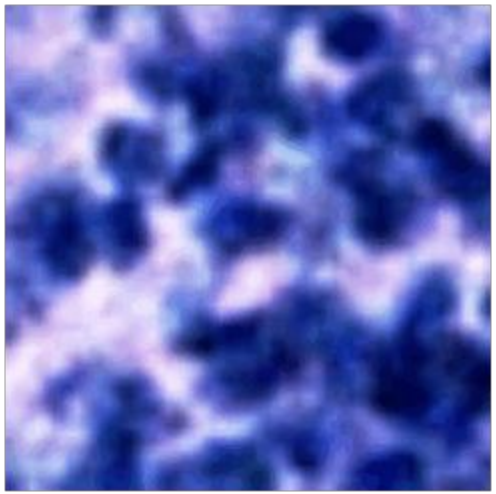
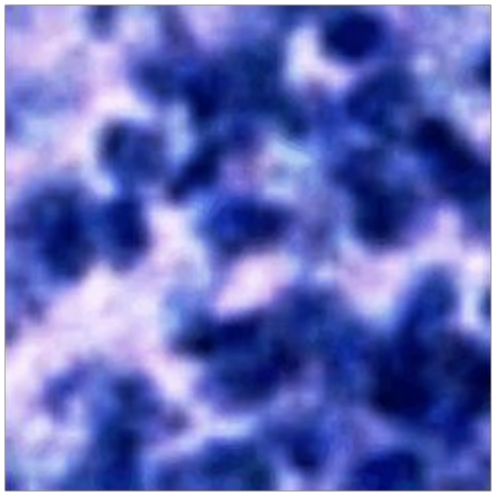
Explaining Generative Encoding and Extensions

David Banh at [AskExplain - interpretable data science]

1. CoreVec

Stage 1
Pre-gcode

$$Y = \alpha X \beta$$

A method that runs a full matrix regression by finding the Solution to AX = B where both A and B are data matrices and X is a matrix of learned values. It is slow due to a large inverse of the matrix A, yet learns a full set of pairwise parameters between features of A and features of B. This pairwise matrix can be considered an adjacency network or complex graph.

Ideas

1. A focus on Complex Networks
    a. as a pairwise adjacency matrix
    b. as learnable parameters in a model
    c. learn a network for both cell and gene interactions

Model History

CoreVec
https://github.com/AskExplain/corevec

Some explanations and slides
https://www.askexplain.com/unifying-models-with-interpretable-parameters

CoreVec
(via Expectation Maximisation)

- Learns full rotations and translations rather than transformations or projections
- Parameters are matrices where scalar elements are pairwise adjacency weights
- Parameters can represent complex networks
- Large number of parameters indicates problem of identifiability
- Similar to Procrustes analysis

Gene-Gene Network	Alignment and Imputation	Cell-Cell Relationships
Biological interpretability Gene relations explained by [Gene to Gene "networks"]	**Biological interpretability** Borrowing information in [reference X] to share with [experiment Y]	**Biological interpretability** Cell relations explained by [Cell to Cell "interactions"]
Mathematical concept Transformation of [Gene space] from [dataset X to dataset Y]	**Mathematical concept** [Projection of data X] to [data Y]	**Mathematical concept** Transformation of [Cell space] from [dataset X to dataset Y]
α	$\alpha X \beta$	β

$$Y = \alpha X \beta + e$$

Experiment dataset of Batch Y	Learned Parameter	Reference dataset of Batch X	Learned Parameter	Residuals

Stage 1A
gcode / gcproc

1. Generative Encoding via Generalised Canonical Procrustes

$$\delta Y \gamma = \alpha X \beta$$

A method that runs a reduced matrix regression by finding the solution to YAX = WBV where A and B are data matrices and Y,X,W,V are matrices of learned values but not of full dimensions of the features of either A or B. X and V or Y and W can be chosen such that the dimensions are reduced and encoded. Inverting V or X can reexpress the full pairwise matrix that can be considered an adjacency network or complex graph.

Ideas

1. Considerations of Algorithmic and Numerical Optimisation
    a. to learn the model faster
    b. make the model identifiable (Occam's razor supports simpler models)
    c. make the model more interpretable

2. Options and Decisions of Model design and choices
    a. more identifiability through sharing parameters
    b. a common latent space to align datasets
    c. brought forward the idea of encoding information
    d. focus on invariance as a way to improve model

3. Connections to other methods
    a. Invariance and deep learning
    b. Signals and gabor filters
    c. Canonical Correlation Analysis
    d. Procrustes Analysis
    e. Singular Value Decomposition
    f. Linear Models
    g. Autoencoders

Current Model

gcode
https://github.com/AskExplain/gcode

biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Summary Sampling
https://github.com/AskExplain/summary_sampling_via_folding

arxiv
https://arxiv.org/abs/2201.08233

gcode/gcproc

(via Variational Inference and Coordinate Ascent)

- Transforms all datasets

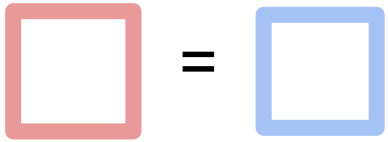
  (X, Y)

  into the same space of lower dimensions via projections

  (K, J, v, u)

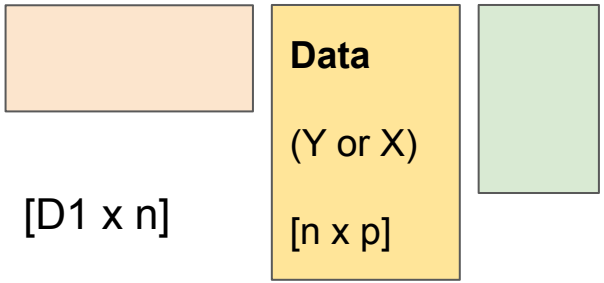
- Model is identifiable now that the matrices have reduced rank (matrix parameters have fewer columns or rows)

$$K\,Y\,v = J\,X\,u$$



D1 x D2

Sample Parameter



[D1 x n]

Feature parameter

[p x D2]

Current Model

gcode/gcproc
(via Variational Inference and Coordinate Ascent)

gcode
https://github.com/AskExplain/gcode

biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Summary Sampling
https://github.com/AskExplain/summary_sampling_via_folding

arxiv
https://arxiv.org/abs/2201.08233

$$Y_{decomposed} = K^T Q v^T \qquad X_{decomposed} = J^T R u^T$$

$$KY_{decomp} v = JX_{decomp} u$$

$$K (K^T Q v^T) v = J (J^T R u^T) u$$

$$KK^T = I \qquad v^T v = I$$

$$Q = R$$

$$JJ^T = I \qquad u^T u = I$$

Current Model

gcode
https://github.com/AskExplain/gcode

biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Summary Sampling
https://github.com/AskExplain/summary_sampling_via_folding

arxiv
https://arxiv.org/abs/2201.08233

gcode/gcproc
(via Variational Inference and Coordinate Ascent)

- Focuses on same properties that makes gcproc flexible
  - Identifiability
  - Interpretability
  - Learnable network
  - Fast learning algorithm
  - Fast transforms

- Includes identical applications that makes gcproc excel
  - Dimension reduction
  - Composable projections
  - Summary Sampling
  - Transferring parameters

Invariance and link to deep learning
e.g. 95% accuracy on MNIST
The model learns (pseudo)invariant parameters by learning to reweight features and samples simultaneously.

Summary Sampling
By encoding the samples into a reduced size, can run a linear regression that gives similar estimated predictions as a model run on the full samples.

Fast Dimension Reduction and Transformation
From a result of clever model design, it is possible to take the inverse of the encoding matrix, rather than the full feature matrix. This helps for both dimension reduction and transformation.

Fast learning algorithm
Decomposing each dataset into parameters such that every inverse operation works on matrices that have been dimensionally reduced. This reduces the runtime by several orders of magnitude.

Current Model

gcode
https://github.com/AskExplain/gcode

biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Summary Sampling
https://github.com/AskExplain/summary_sampling_via_folding

arxiv
https://arxiv.org/abs/2201.08233

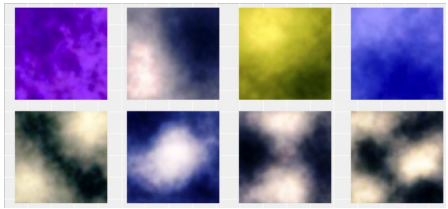
Invariance and link to deep learning
e.g. 95% accuracy on MNIST

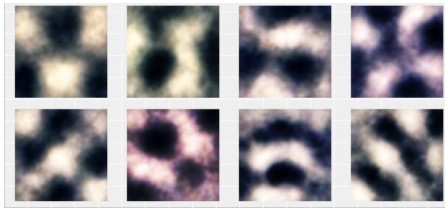
The feature encoder is a learnable matrix parameter where columns are components.

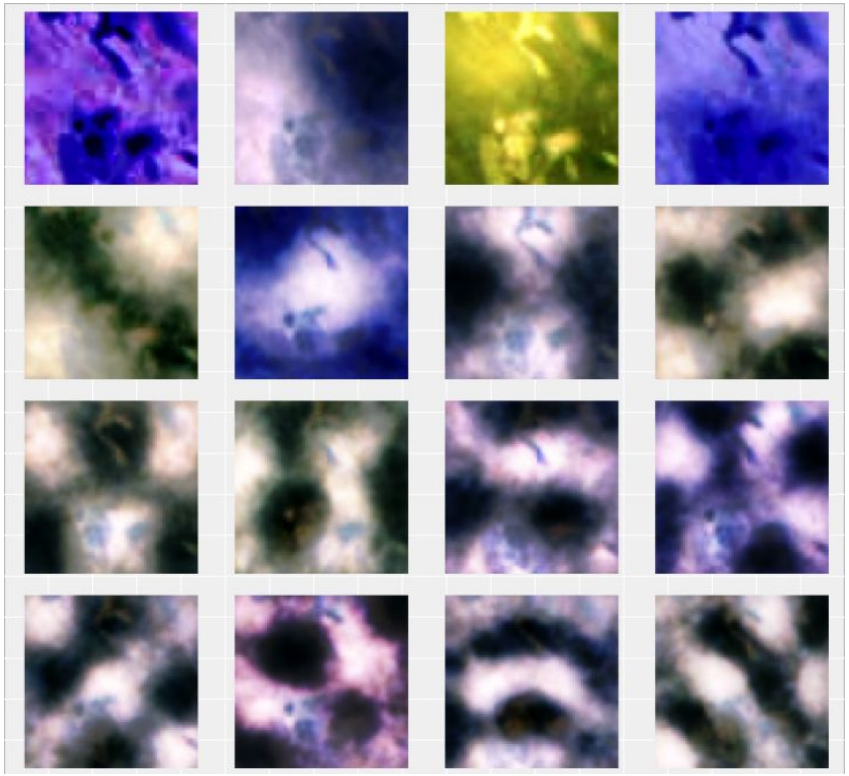
To encode a dataset, observed data matrix is matrix multiplied by the feature parameter.

When analysing images, components act similar to filters.

Generative Encoders: visualising components of the feature encoder that operate as filters for cell shapes in Spatial Transcriptomics







Current Model

gcode
https://github.com/AskExplain/gcode

biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Summary Sampling
https://github.com/AskExplain/summary_sampling_via_folding

arxiv
https://arxiv.org/abs/2201.08233

Fast dimension reduction and transformation

The feature encoder reduces X to a D dimensional feature space.

The sample encoder reduces X to a D dimensional sample space

These operations are via a straightforward matrix multiplication.

For transformations, notice in the very last expression Y and X are related via a projection from u and v.

Dimension Reduction

$$Feature\ encoding = Xu$$
$$(u\ is\ feature \times D_1)$$
$$Sample\ encoding = KX$$
$$(K\ is\ D_1 \times sample)$$

Transformation

$$Assume\ K\ is\ common\ across\ Y\ and\ X:$$
$$Y = K^T Q v^T$$
$$X = K^T Q u^T$$

$$Y = K^T Q v^T = Xu(u^T u)^{-1} v^T$$

$$((u^T u)^{-1}\ is\ D_1 \times D_1)$$

Current Model

gcode
https://github.com/AskExplain/gcode

biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Summary Sampling
https://github.com/AskExplain/summary_sampling_via_folding

arxiv
https://arxiv.org/abs/2201.08233

Fast learning algorithm

Decomposing each dataset into parameters such that every inverse operation works on matrices that have been dimensionally reduced. This reduces the runtime by several orders of magnitude.

Notice in the expression on the right, the positive definite square matrices are of D dimensions, where D is the number of reduced components (usually set somewhere between 2 and 100).

Thus the limiting computational step is not the inversion of a D x D matrix, but matrix multiplication when there are millions of cells and hundreds of thousands of features.

$$X_{decomposed} = J^T R u^T$$

$$J^T = X (Ru^T)^T ((Ru^T)(Ru^T))^{-1}$$

$$Ru^T : D_1 \times features$$

$$(Ru^T)(Ru^T) : D_1 \times D_1$$

$$D_1 = number\ of\ components$$

$$(eg.\ D_1 = 30)$$

$$u^T = ((J^T R)^T (J^T R))^{-1} (J^T R) X$$

$$J^T R : samples \times D_2$$

$$(J^T R)^T (J^T R) : D_2 \times D_2$$

$$D_2 = number\ of\ components$$

$$(eg.\ D_2 = 30)$$

Stage 1B

Summary
Sampling
with
gcode /
gcproc

$$\|Y - f(X, \theta)\|_2^2$$

$$\|\alpha Y - f(\alpha X)\|_2^2 = \|g(Y) - f(g(X))\|_2^2$$

$$\left\|g^{-1}(g(Y)) - g^{-1}(f(Z\beta^T, \theta))\right\|_2^2 = \left\|Y - g^{-1}(f(Z\beta^T, \theta))\right\|_2^2 = \left\|Y - \hat{Y}\right\|_2^2$$

$$\left\|Y - \hat{Y}\right\|_2^2 = \left\|Y - f(g^{-1}(Z\beta^T), \theta)\right\|_2^2 = \left\|Y - f(\hat{X}, \theta)\right\|_2^2$$

The last line follows from the estimate of X, given as $\hat{X} = g^{-1}(Z\beta^T) = \alpha^T Z\beta^T$

Ideas

Summarising or Folding the samples reduces the total number of samples to a manageable amount to run on modern day machines.

The idea supposes that when there are many samples, some samples cluster and thus share more relevant information - making them nearer neighbours. The concept of summarising concentrates the sample information, akin to folding a piece of paper to reduce the total area. Running an algorithm on the folded samples then gives an [output] in the reduced sample space. The next step is to unfold the [output] to match the structure of the full samples in the original data.

Property of Model

Summary Sampling
https://github.com/AskExplain/summary_sampling_via_folding

arxiv
https://arxiv.org/abs/2201.08233

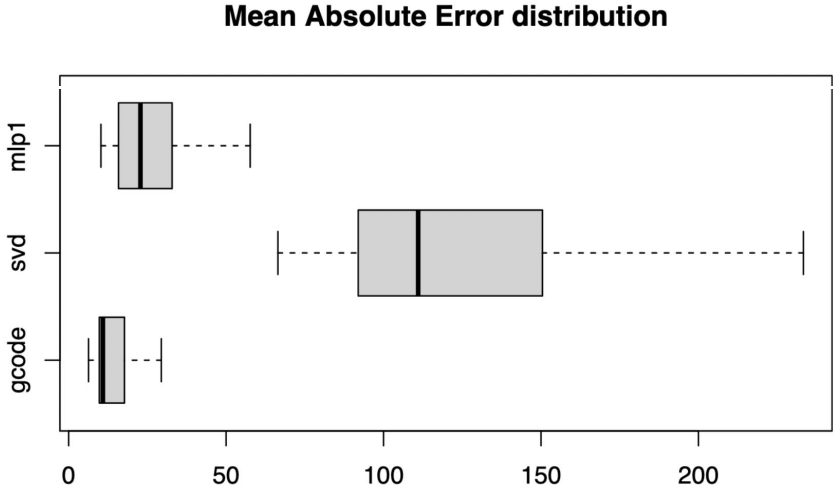
Summary Sampling

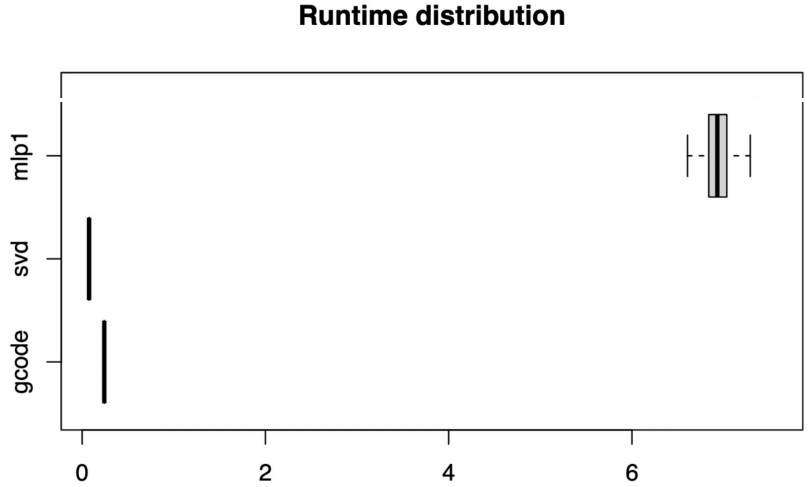
By encoding the samples int a reduced size, can run any regression algorithm that gives estimated predictions similar to a model run on the full samples.

Metrics to support similarity between full samples and summarised samples are RMSE and pearson correlation.

Preprint on arXiv shows evidence on linear regression and random forest regression.

Would like to try reducing sample sizes in deep learning pipelines on GPUs to speed up runtime and improve efficiency while retaining accuracy.





Stage 3
GeneCodeR

GeneCodeR:

Histology adaptation through perturbing expected gene expression levels

Ideas

Biology Perturbation

1. Can the method be extended to learn on spatial data samples that represent the developmental trajectory of tissues and cells?
2. Could a platform be created that projects how a tissue sample can develop over time (from an organoid state, to a more mature, differentiated state) based on cell modality perturbations?
3. Is it possible to improve knowledge on single gene functions or multiple interactions and extend databases through the concept of in-silico perturbations?
4. Could this build on other areas of research, for example with modality maps used on in-silico perturbations of genetic variants to finely map their influence on gene expression and spatial tissue?

Machine learning Performance

5. Can Generative Encoding be used as a platform to create training data to **augment** deep learning method and improve performance?
6. Can **Summary Sampling** via Generative Encoding speed up the performance of machine learning algorithms when learning on images?

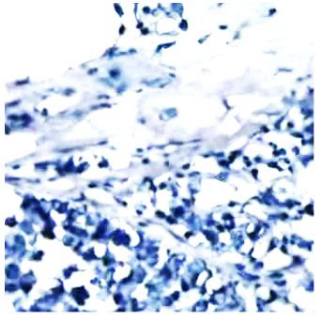
Next Model

GeneCodeR
https://github.com/AskExplain/GeneCodeR

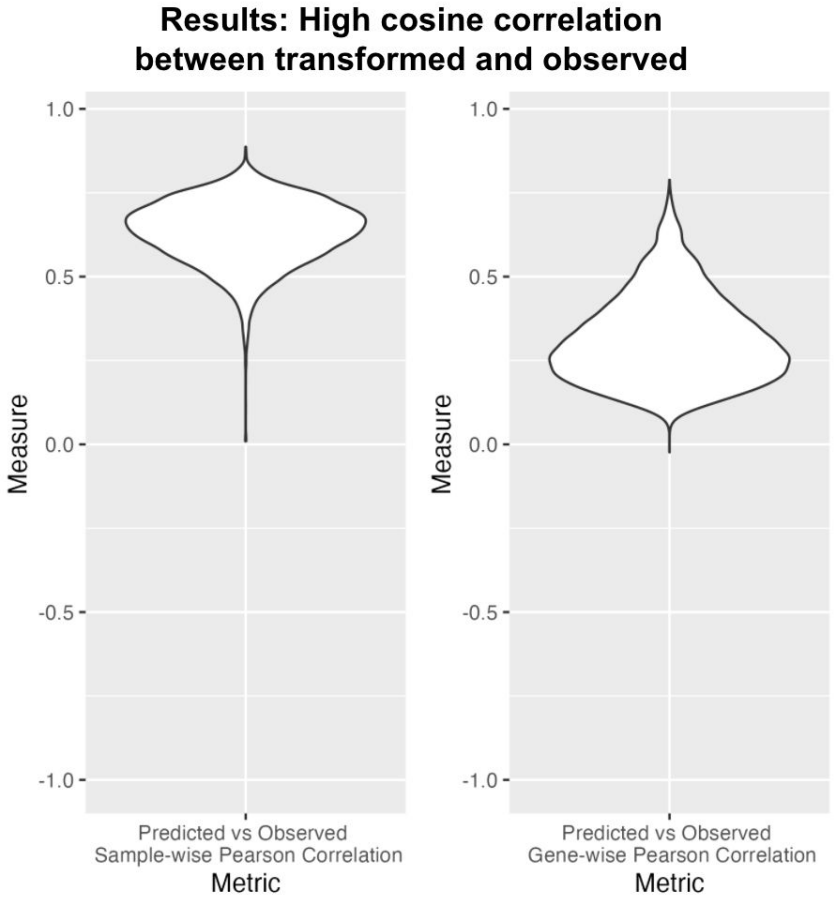
GeneCodeR Documentation
https://askexplain.github.io/GeneCodeR/articles/analysis_of_spatial_transcriptomics.html

biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Base relational equivalence



GEX
Transformed
vs
Observed



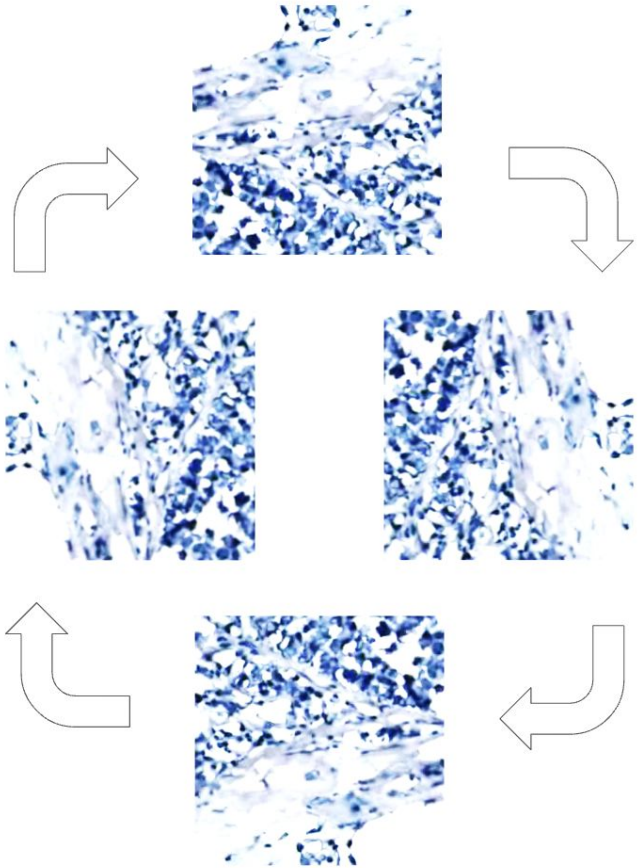
Next Model

GeneCodeR
https://github.com/AskExplain/GeneCodeR

GeneCodeR Documentation
https://askexplain.github.io/GeneCodeR/articles/analysis_of_spatial_transcriptomics.html

biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Spatial Rotation



Results: No statistically significant difference across rotations

Rotation comparison	P-value (non-adjusted)
0 vs 90 degrees	0.76472
0 vs 180 degrees	0.96699
0 vs 270 degrees	0.81952

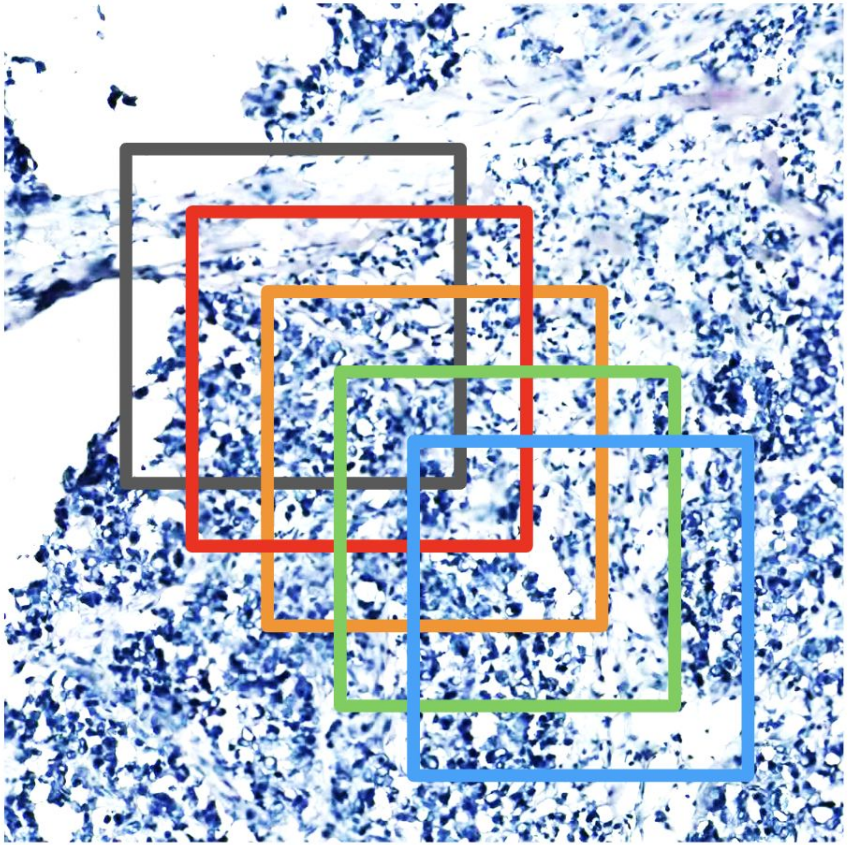
Next Model

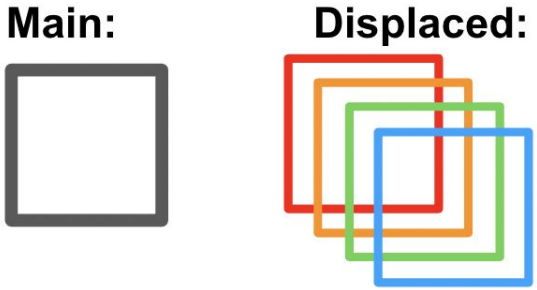
GeneCodeR
https://github.com/AskExplain/GeneCodeR

GeneCodeR Documentation
https://askexplain.github.io/GeneCodeR/articles/analysis_of_spatial_transcriptomics.html

biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Spatial Displacement





Results: Increasing statistical significant difference with displacement

Displacement comparison	P-value (non-adjusted)
0 vs 2 displaced pixels	0.63127
0 vs 4 displaced pixels	0.30681
0 vs 8 displaced pixels	0.04463

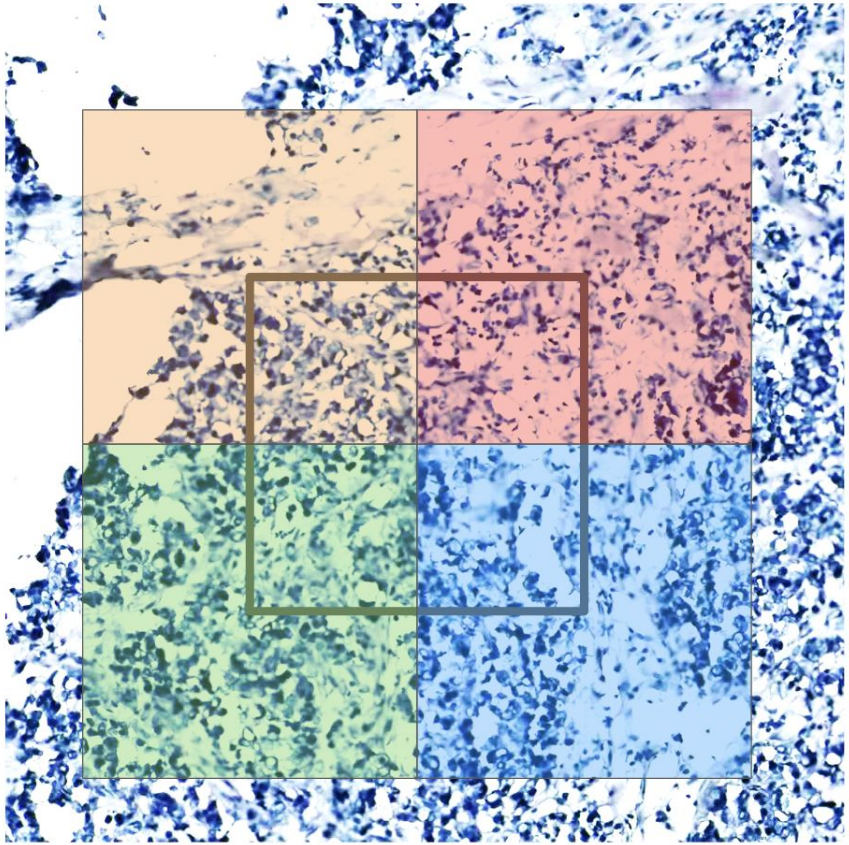
Next Model

GeneCodeR
https://github.com/AskExplain/GeneCodeR

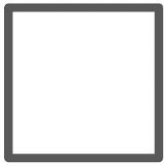
GeneCodeR Documentation
https://askexplain.github.io/GeneCodeR/articles/analysis_of_spatial_transcriptomics.html

biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

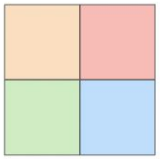
Spatial Similarity



Main:



Similar:



Results: No statistical significant difference across spatially similar spots

Similarity comparison	P-value (non-adjusted)
0,0 vs -3,-3 x,y coord displace	0.61131
0,0 vs 3,-3 x,y coord displace	0.5161
0,0 vs -3,3 x,y coord displace	0.21426
0,0 vs 3,3 x,y coord displace	0.61131

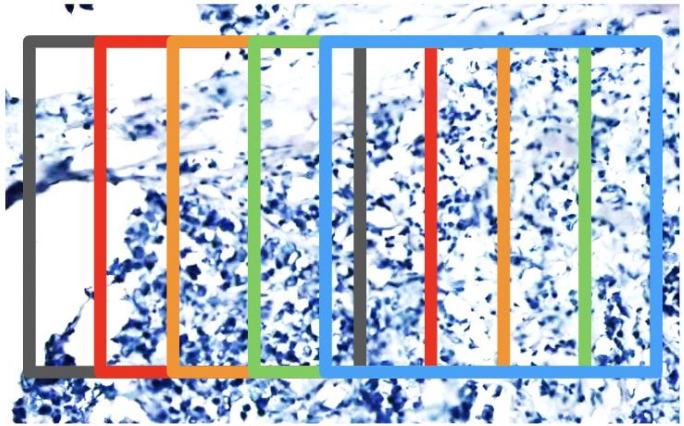
Next Model

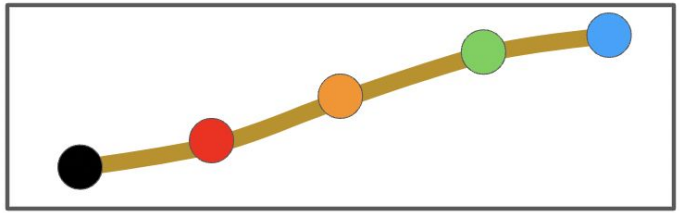
GeneCodeR
https://github.com/AskExplain/GeneCodeR

GeneCodeR Documentation
https://askexplain.github.io/GeneCodeR/articles/analysis_of_spatial_transcriptomics.html

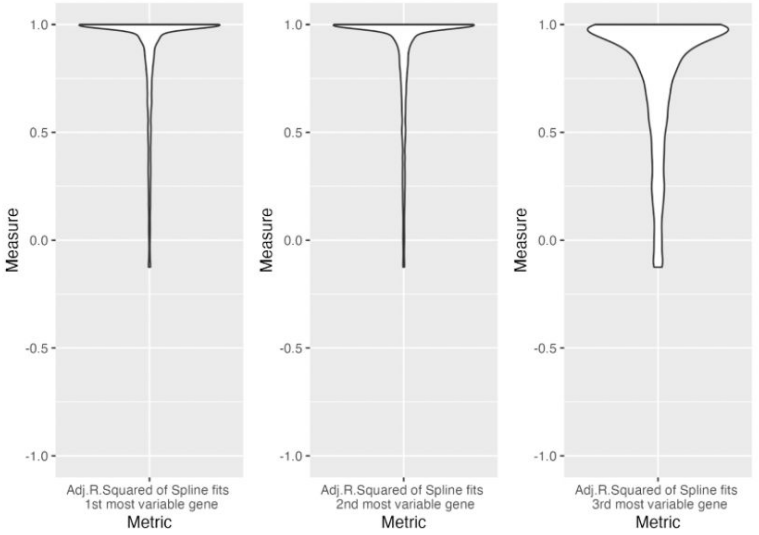
biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Spatial Signal

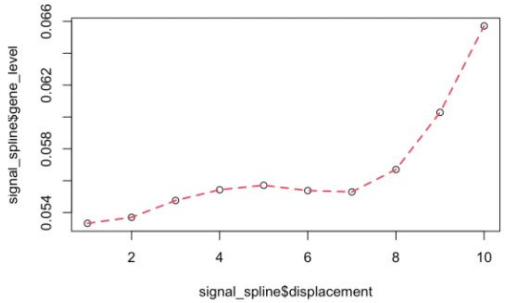




Results: Spline fits well as displacement is incremented



Adj.R-squared: 0.844571155864712



Next Model

GeneCodeR
https://github.com/AskExplain/GeneCodeR

GeneCodeR Documentation
https://askexplain.github.io/GeneCodeR/articles/analysis_of_spatial_transcriptomics.html

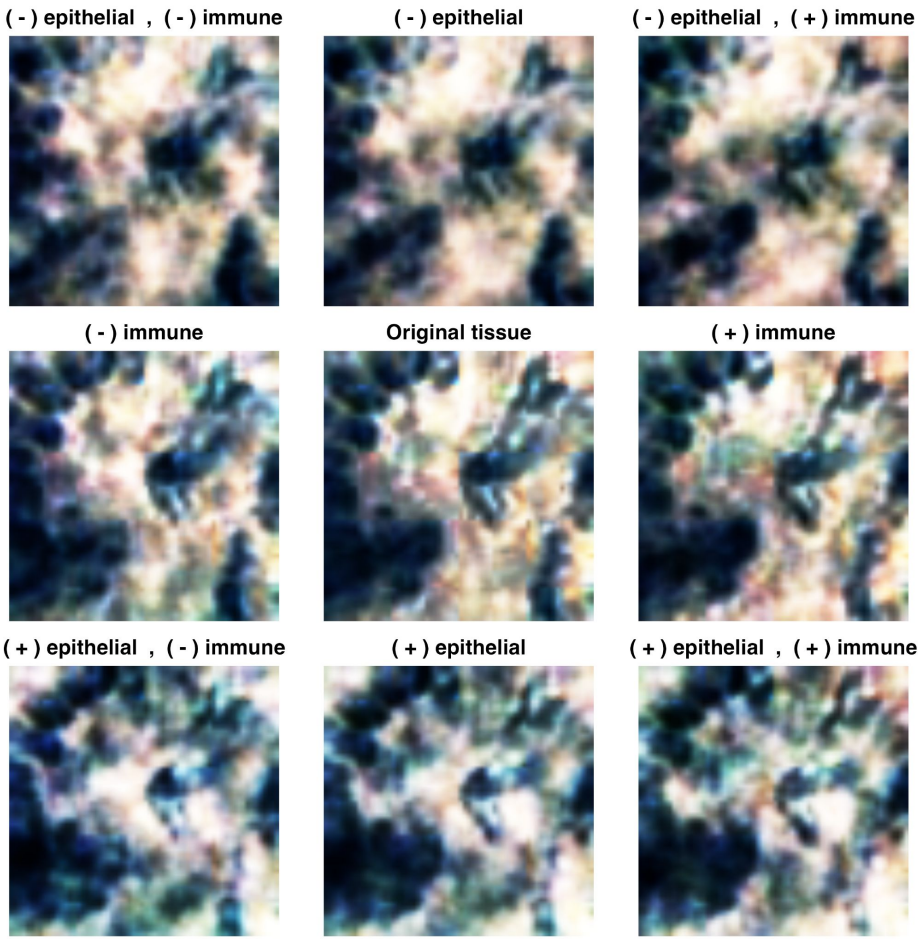
biorxiv
https://www.biorxiv.org/content/10.1101/2021.07.09.451779v2

Machine Learning Performance

Summary sampling described previously can be used to speed up and enhance the learning capabilities of neural networks

Can histology perturbation be used to improve deep learning training via augmentation?

Given an almost infinite possible number of images derived from perturbing gene expression, can deep learning performance improve?



Stage 4
lmform

Linear Mixed Forms (lmform):

$$Y\alpha = X\beta + Zu + e$$

Linear Mixed Models with encoding transformations

Ideas

1. Given the success of linear mixed models in genomics, can the same properties and qualities of the model be translated to Spatial Transcriptomics?
2. This can consider covariates much more flexibly, and random effects in addition to fixed effects

Newest Model

lmform
https://github.com/AskExplain/lmform

Linear Mixed Forms

Ideas are shared from Generative Encoding such that:

- each dataset (Y, X, Z) can be decomposed into a latent code (K or H, or K+H) and a parameter (alpha, beta, or u)

- Fixed effects are from X.beta
- Random effects from Z.u

- K is the latent fixed effect
- H is the latent random effect

$$Y\alpha = X\beta + Zu + e$$

$$Y = (K + H)\alpha^T$$

$$X = K\beta^T$$

$$Z = Hu^T$$

Newest Model

lmform
https://github.com/AskExplain/lmform

Linear Mixed Forms

Extending the concept of latent components to random and fixed effects:

- K is the latent fixed effect
- H is the latent random effect

- Y is now fully decomposable into a latent random effect and a fixed effect
- In standard linear mixed models, the latent code does not exist

- *Design of algorithm is similar to gcode and benefits from speed and simplicity*

$$Substituting...$$

$$(K + H)\alpha^T\alpha = K\beta^T\beta + Hu^Tu + e$$

$$Rearranging...$$

$$K\alpha^T\alpha + H\alpha^T\alpha = K\beta^T\beta + Hu^Tu + e$$

$$Fixed\ effect\ expression...$$

$$K\alpha^T\alpha = K\beta^T\beta$$

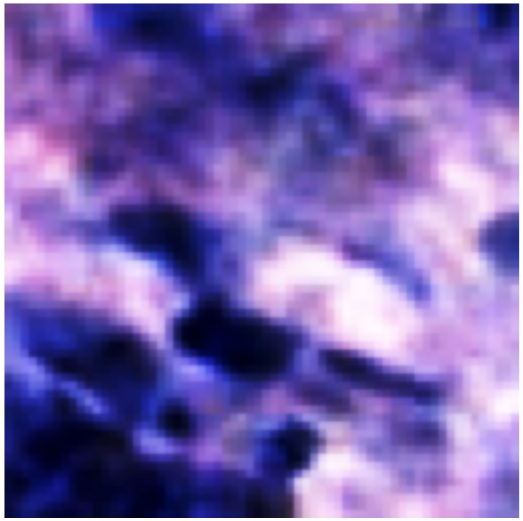
$$Random\ effect\ expression...$$

$$H\alpha^T\alpha = Hu^Tu + e$$

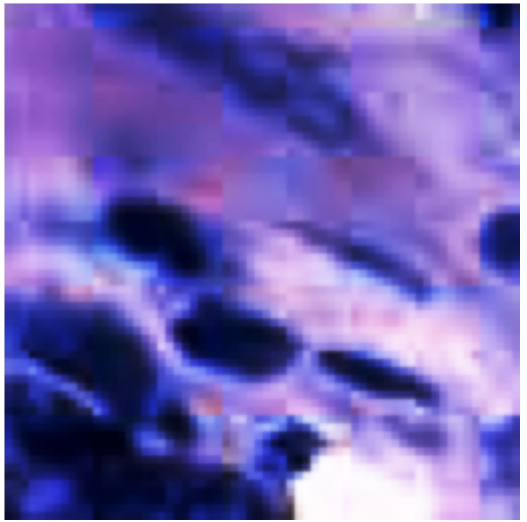
Newest Model

lmform
https://github.com/AskExplain/lmform

Gene expression (***) transformation to histology



Real histology unperturbed direct from image



Representation of cells clearly reflects structure and shape

Background colour different due to being more uniform and larger

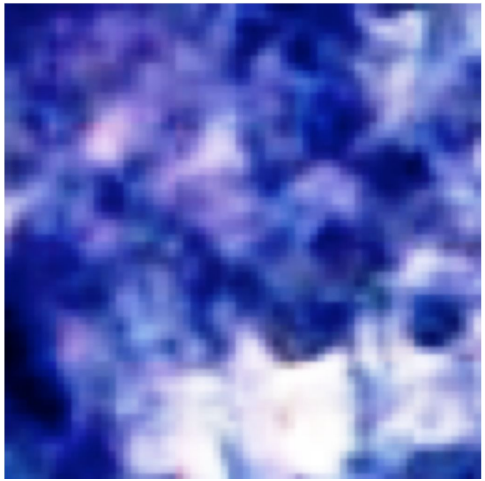
Details are much finer than Generative Encoders

(***) Gene expression taken on **TRAIN** set

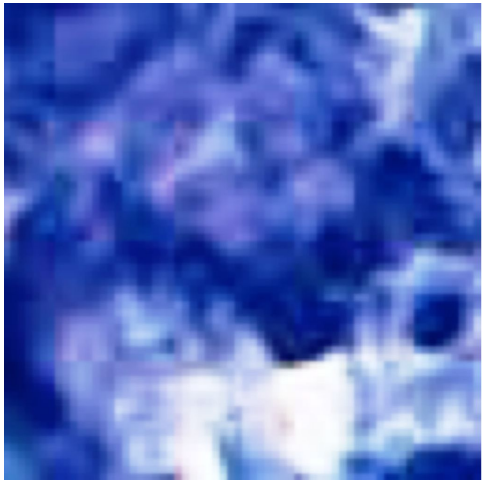
Newest Model

lmform
https://github.com/AskExplain/lmform

Gene expression (***) transformation to histology



Real histology unperturbed direct from image



Representation of cells clearly reflects structure and shape

Background colour different due to being more uniform and larger

Details are much finer than Generative Encoders

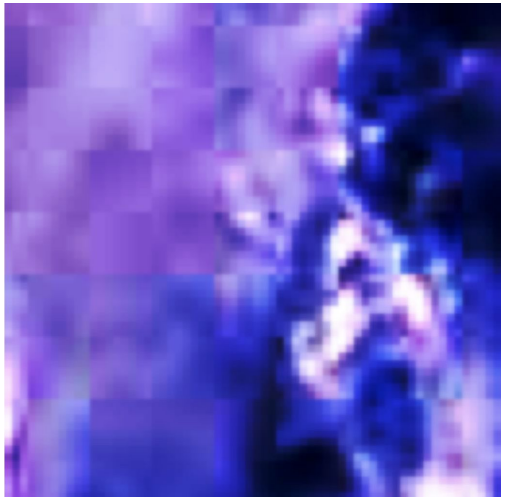
(***) Gene expression taken on **TRAIN** set

Newest Model

lmform
https://github.com/AskExplain/lmform

Gene expression (***) transformation to histology

Real histology unperturbed direct from image



Representation of cells clearly reflects structure and shape

Background colour different due to being more uniform and larger

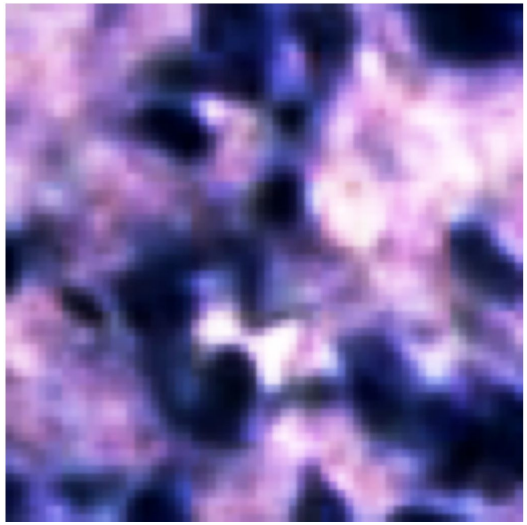
Details are much finer than Generative Encoders

(***) Gene expression taken on **TRAIN** set

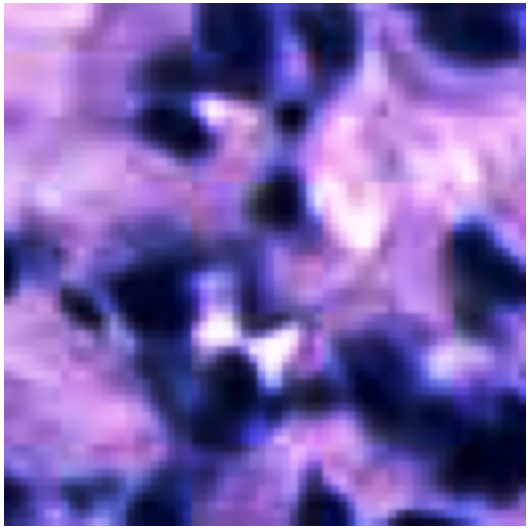
Newest Model

lmform
https://github.com/AskExplain/lmform

Gene expression (***) transformation to histology



Real histology unperturbed direct from image



Representation of cells clearly reflects structure and shape

Background colour different due to being more uniform and larger

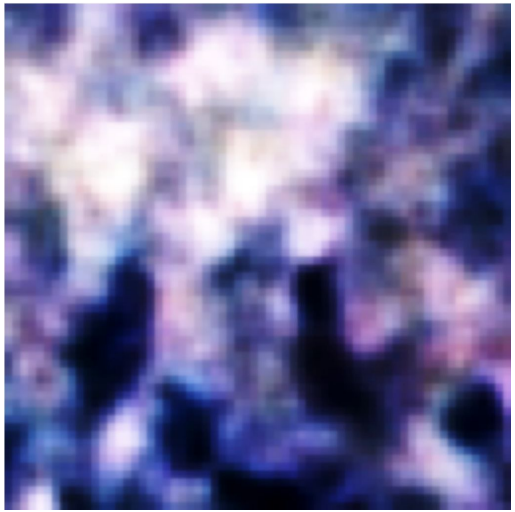
Details are much finer than Generative Encoders

(***) Gene expression taken on **TRAIN** set

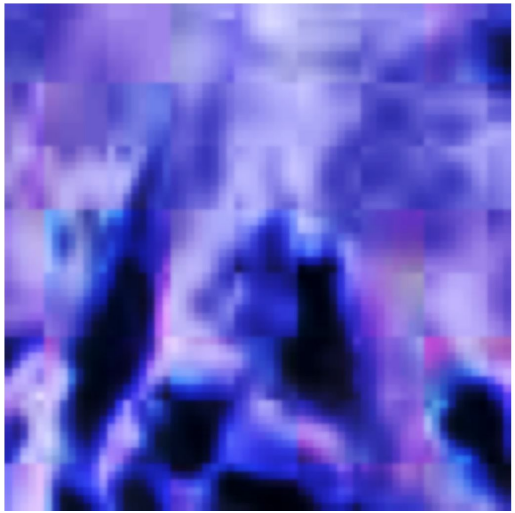
Newest Model

lmform
https://github.com/AskExplain/lmform

Gene expression (***) transformation to histology



Real histology unperturbed direct from image



Representation of cells clearly reflects structure and shape

Background colour different due to being more uniform and larger

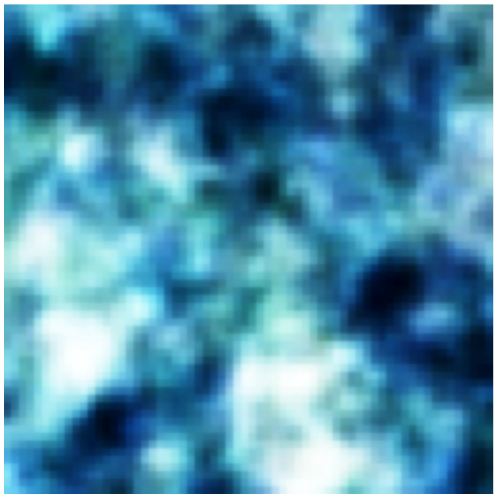
Details are much finer than Generative Encoders

(***) Gene expression taken on **TRAIN** set

Newest Model

lmform
https://github.com/AskExplain/lmform

Gene expression (***) transformation to histology



Real histology unperturbed direct from image

Focus on:

- Colour of extracellular matrix
- Number of cells
- Cell size relative to total image and other cells

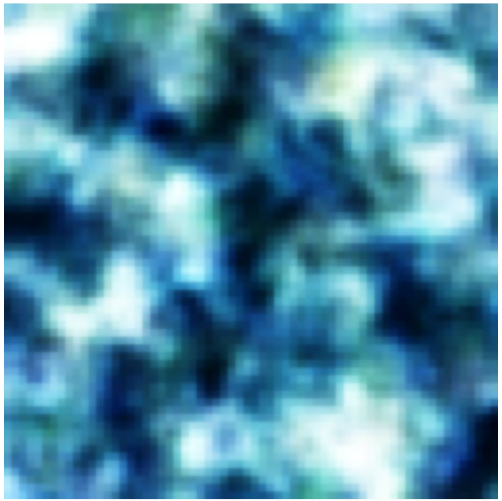
Details of structure, colour and shape are consistently better than Generative Encoders

(***) Gene expression taken on **TEST** set

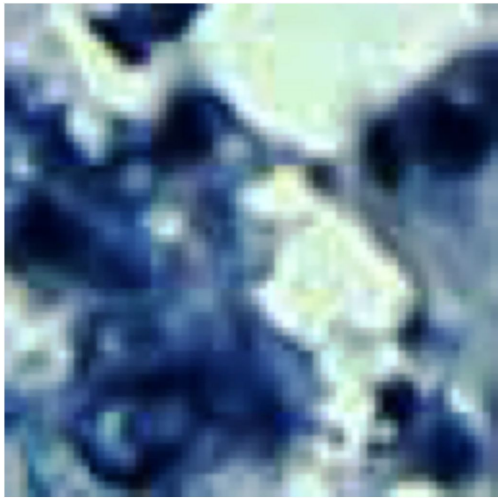
Newest Model

lmform
https://github.com/AskExplain/lmform

Gene expression (***) transformation to histology



Real histology unperturbed direct from image



Focus on:

- Colour, density and coverage of extracellular matrix
- Number of cells
- Cell size relative to total image and other cells

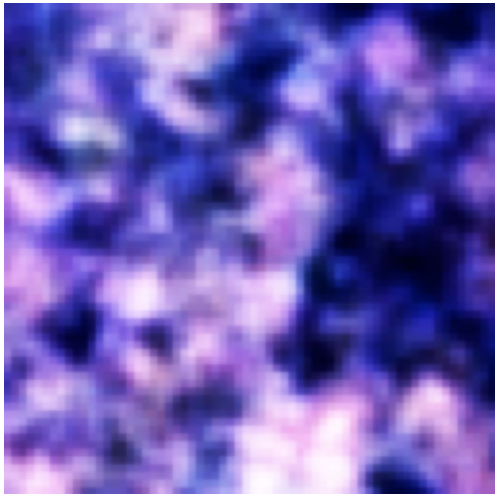
Details of structure, colour and shape are consistently better than Generative Encoders

(***) Gene expression taken on **TEST** set

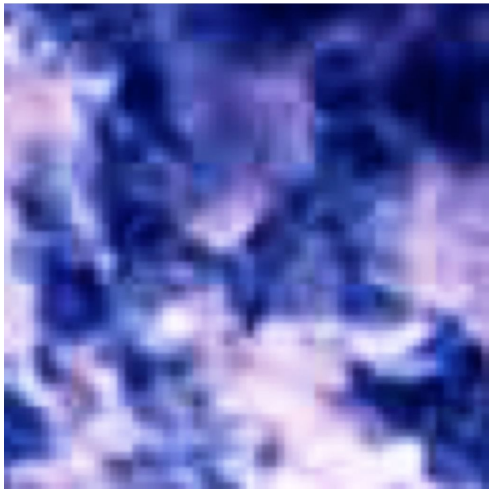
Newest Model

lmform
https://github.com/AskExplain/lmform

Gene expression (***) transformation to histology



Real histology unperturbed direct from image



Focus on:

- Colour of extracellular matrix
- Number of cells
- Cell size relative to total image and other cells

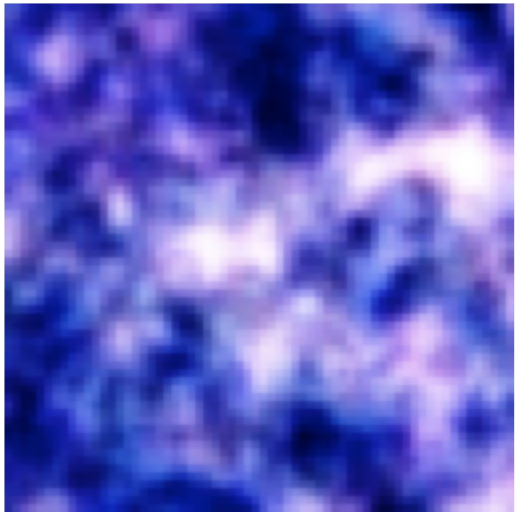
Details of structure, colour and shape are consistently better than Generative Encoders

(***) Gene expression taken on **TEST** set

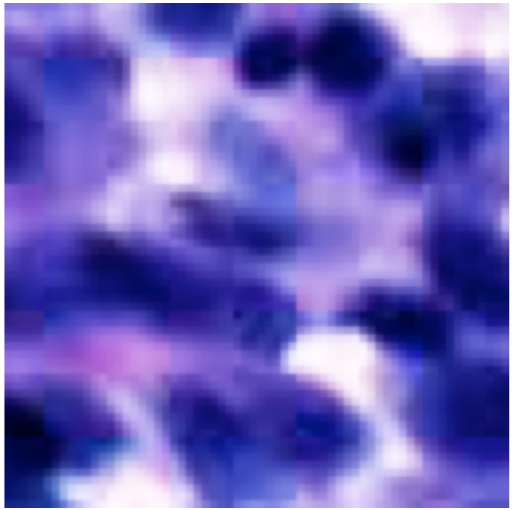
Newest Model

lmform
https://github.com/AskExplain/lmform

Gene expression (***) transformation to histology



Real histology unperturbed direct from image



Focus on:

- Colour, density and coverage of extracellular matrix
- Number of cells
- Cell size relative to total image and other cells

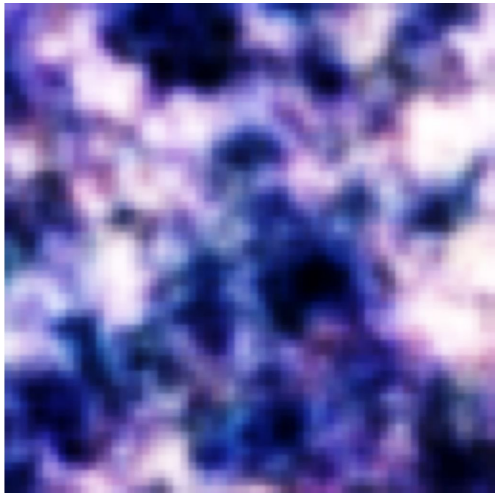
Details of structure, colour and shape are consistently better than Generative Encoders

(***) Gene expression taken on **TEST** set

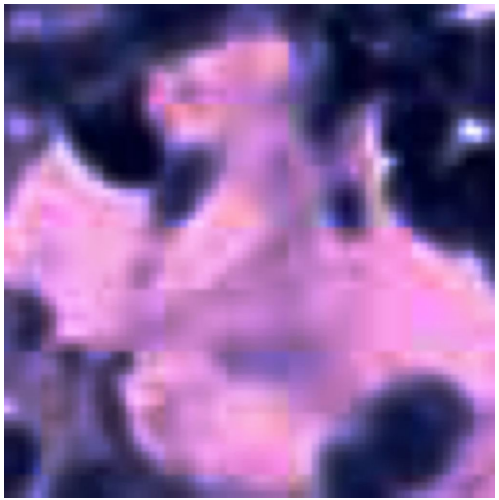
Newest Model

lmform
https://github.com/AskExplain/lmform

Gene expression (***) transformation to histology



Real histology unperturbed direct from image



Focus on:

- Colour, density and coverage of extracellular matrix
- Number of cells
- Cell size relative to total image and other cells

Details of structure, colour and shape are consistently better than Generative Encoders

(***) Gene expression taken on **TEST** set

Summary

1. **CoreVec** - a full rank transformation algorithm focused on networks
2. **gcproc / gcode** - an encoding algorithm focused on transformations
3. **Summary Sampling** - a faster and more efficient way to run regression
4. **GeneCodeR** - a wrapper of gcode to extend the model
5. **lmform** - a more accurate representation of signals in data

Acknowledgements as collaborators

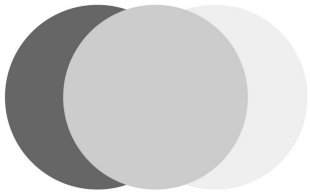
Dr Alan Huang **University of Queensland**

Acknowledgements for inspiration

Cameron Gordon	**University of South Australia**
Ryan Deslandes	**University of Queensland**
Alex Alsaffar	**University of Queensland**
Olivia Ou	**University of Queensland**
Professor Geoffrey McLachlan	**University of Queensland**
Dr Quan Nguyen	**University of Queensland**
Dr Jessica Mar	**University of Queensland**
Dr Ruth Pidsley	**Garvan Institute of Medical Research**

Our vision is to create great collaborations from science to industry

Through an interest and passion for science to progress, our commitment is to work on, communicate around, and support collaborations



AskExplain
ask@askexplain.com