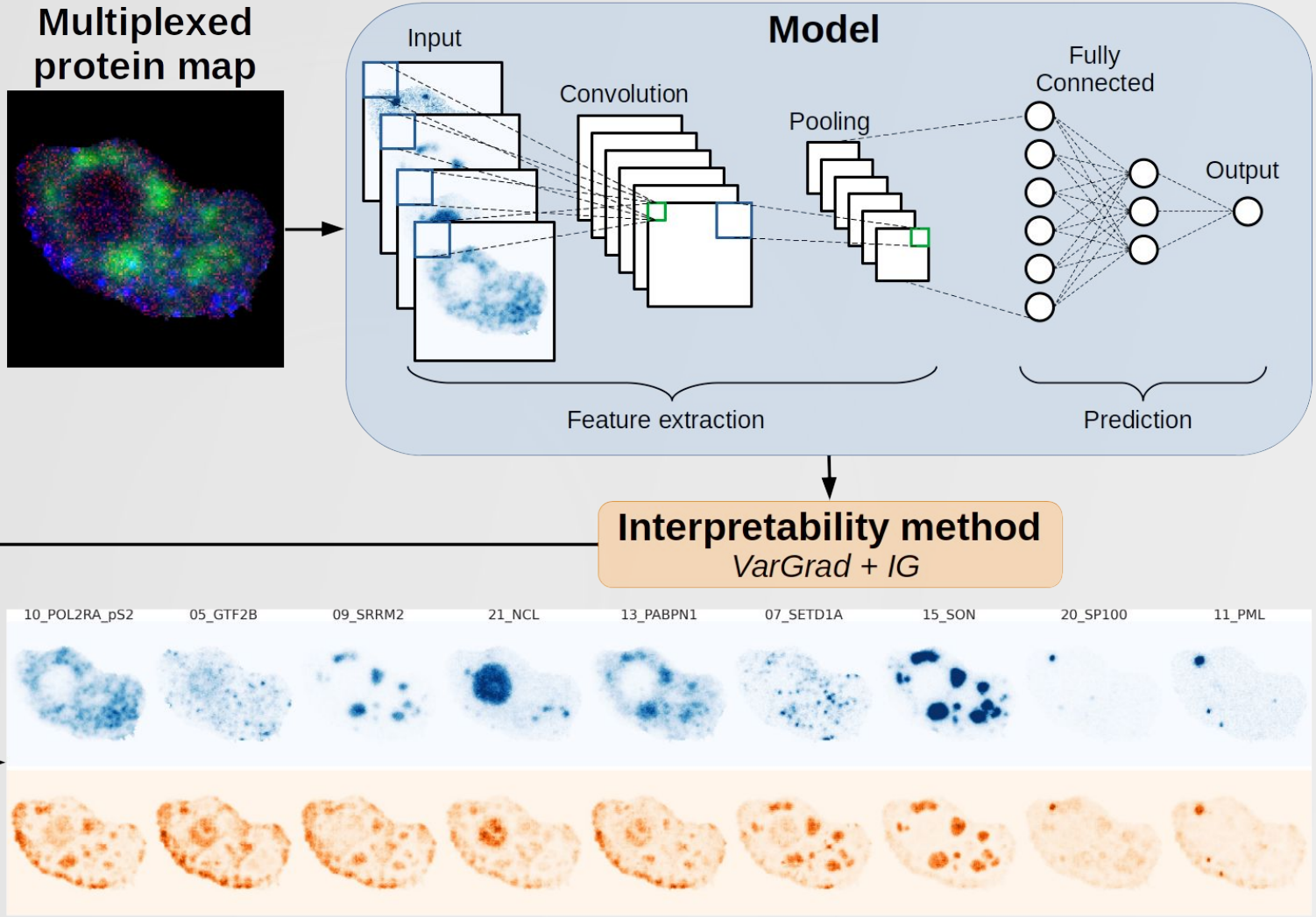
Technische Universität München
Department of Mathematics
Chair of Mathematical Modeling of Biological Systems

HMGU
ICB

Predicting transcription rate from multiplexed protein maps using deep learning

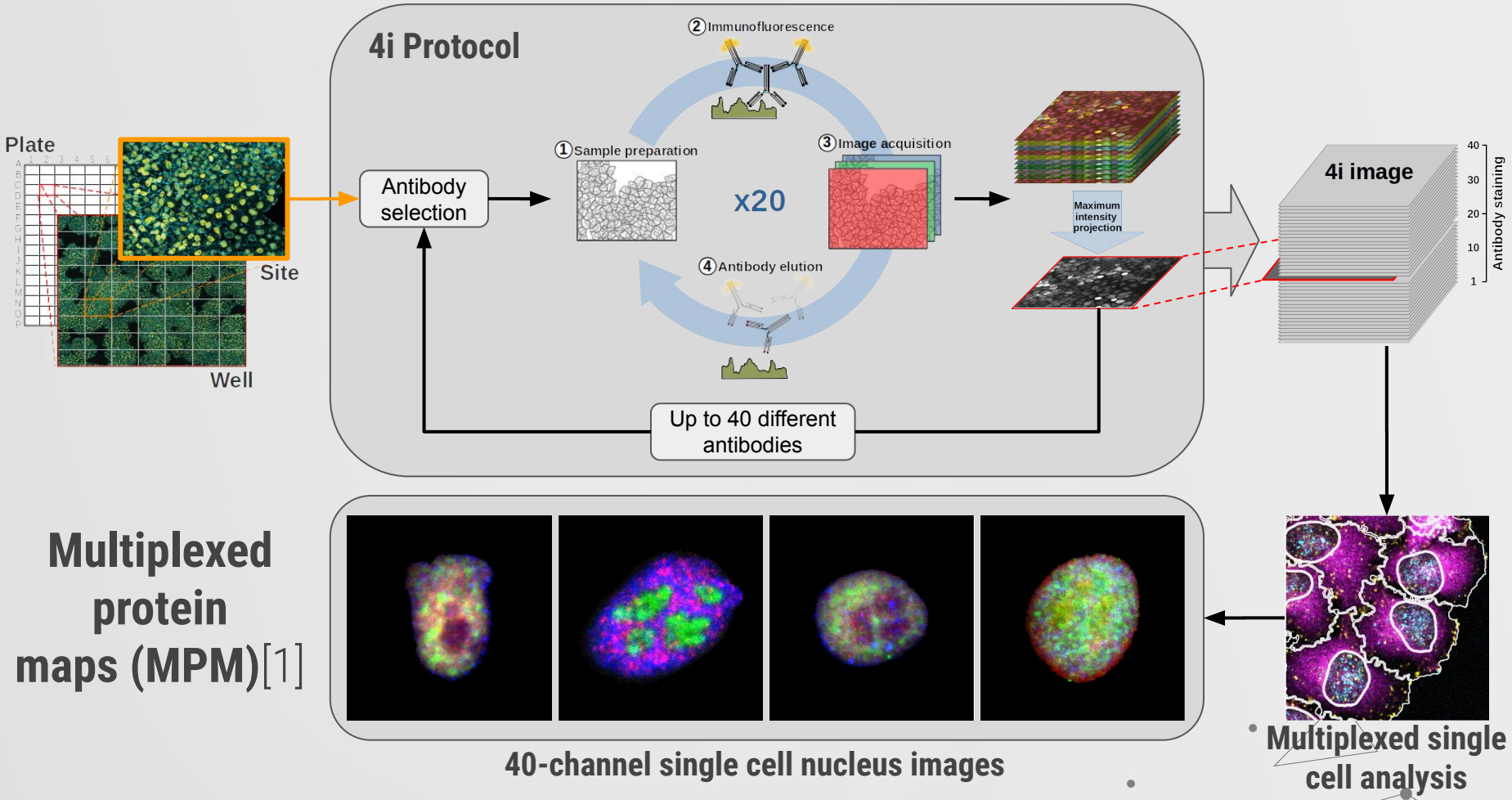
Andres Becker
May 25, 2021
Advisor: Dr. Hannah Spitzer

Road Map


Multiplexed protein map
Model
Input
Convolution
Pooling
Fully Connected
Output
Feature extraction
Prediction
Interpretability method
VarGrad + IG
10_POL2RA_pS2
05_GTF2B
09_SRRM2
21_NCL
13_PABPN1
07_SETD1A
15_SON
20_SP100
11_PML

01

Multiplexed protein maps

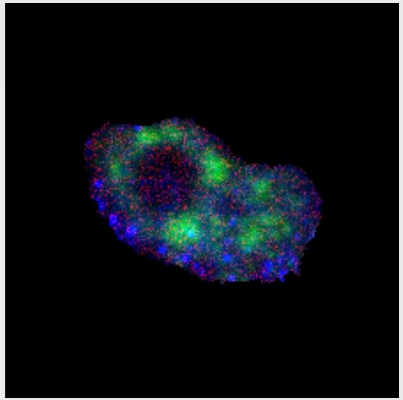


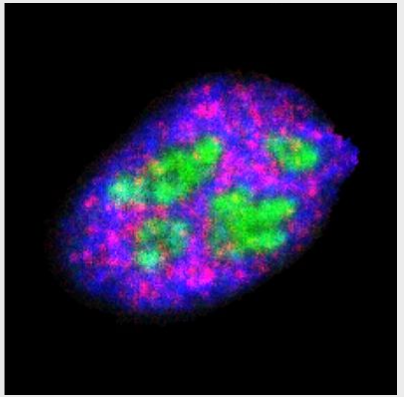
[1] G. Gut, M. D. Herrmann, and L. Pelkmans. "Multiplexed protein maps link subcellular organization to cellular states".

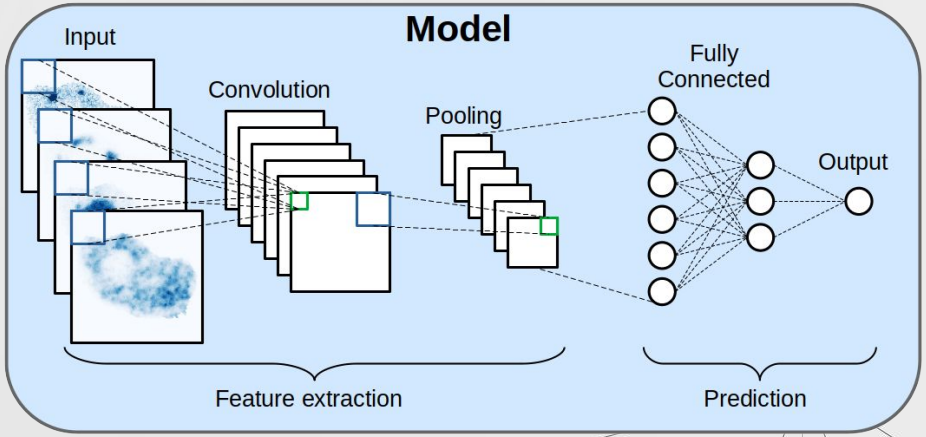
Multiplexed protein maps

The data provided by **Scott Berry** from Pelkmans Lab (at the University of Zurich)

- **3703** images of cell nucleus
- **33** channels (proteins markers)
- Cell **transcription rate** (TR) is used as target variable
    - Average of channel 5-Ethynyl Uridine (EU), which marks **newly transcribed RNA molecules in 30 minutes**







02

Data preprocessing and Data augmentation

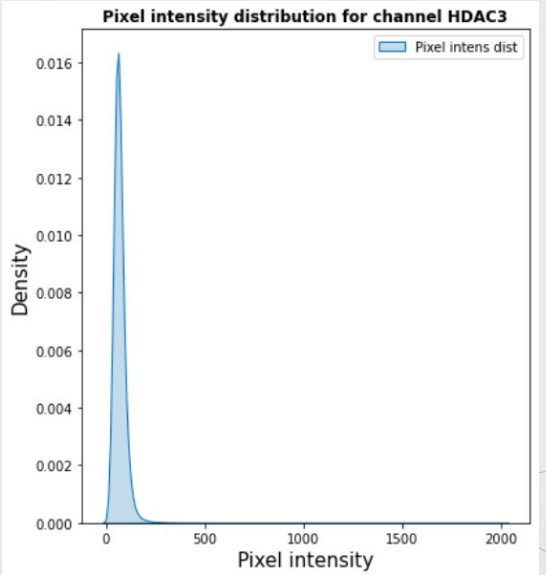
Data preprocessing

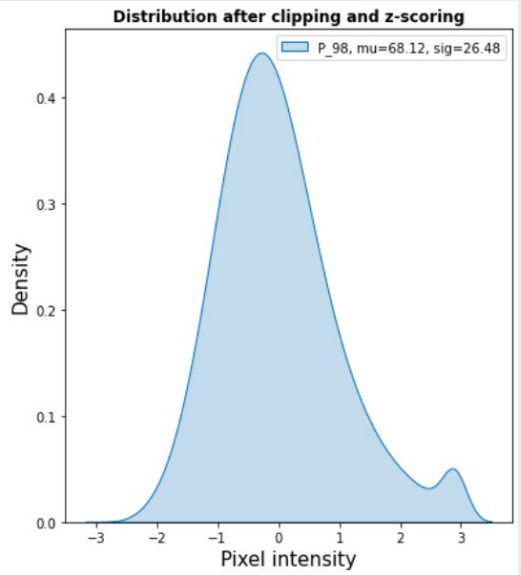
To improves model performance and training stability:

- Per-channel 98% percentile clipping
- Per-channel standardization

$$z_c = \frac{x_c - \mu_c}{\sigma_c}$$

- After preprocessing, the cell images are stored as a TensorFlow dataset

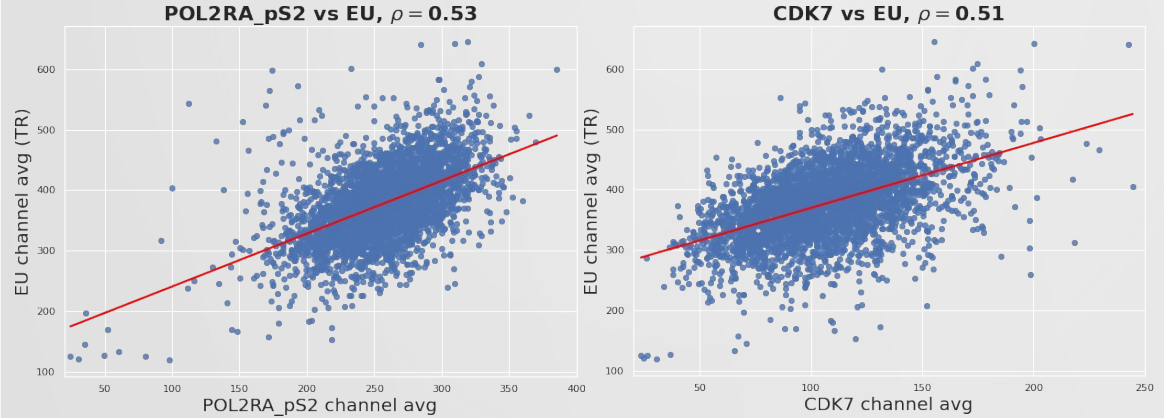


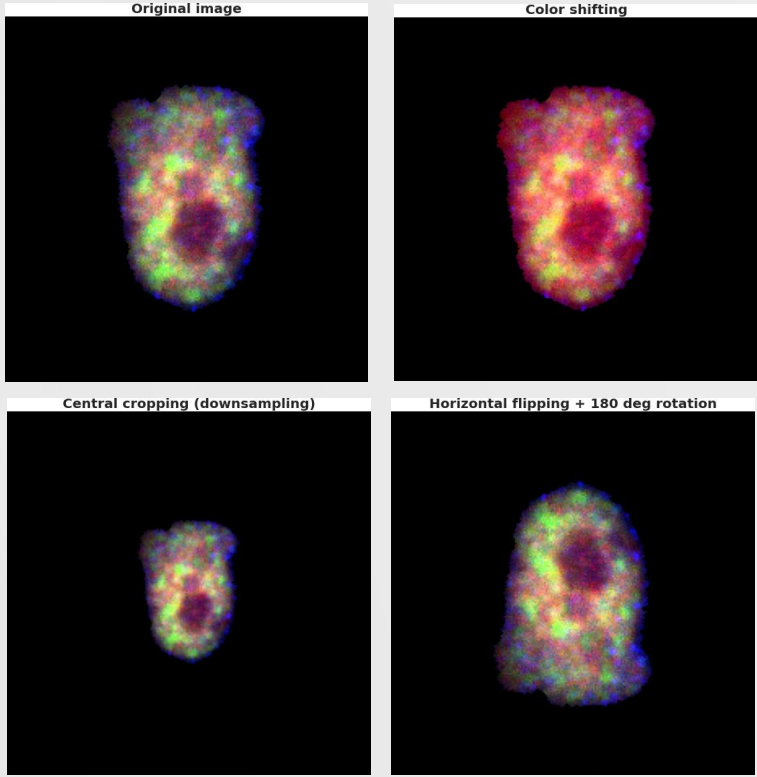


Data augmentation

To alleviate overfitting [1] and encourage the model to focus on spatial information:

- Random color shifting
- Random upsampling and downsampling (center zoom in/out)
- Random horizontal flipping + 90 degree rotations



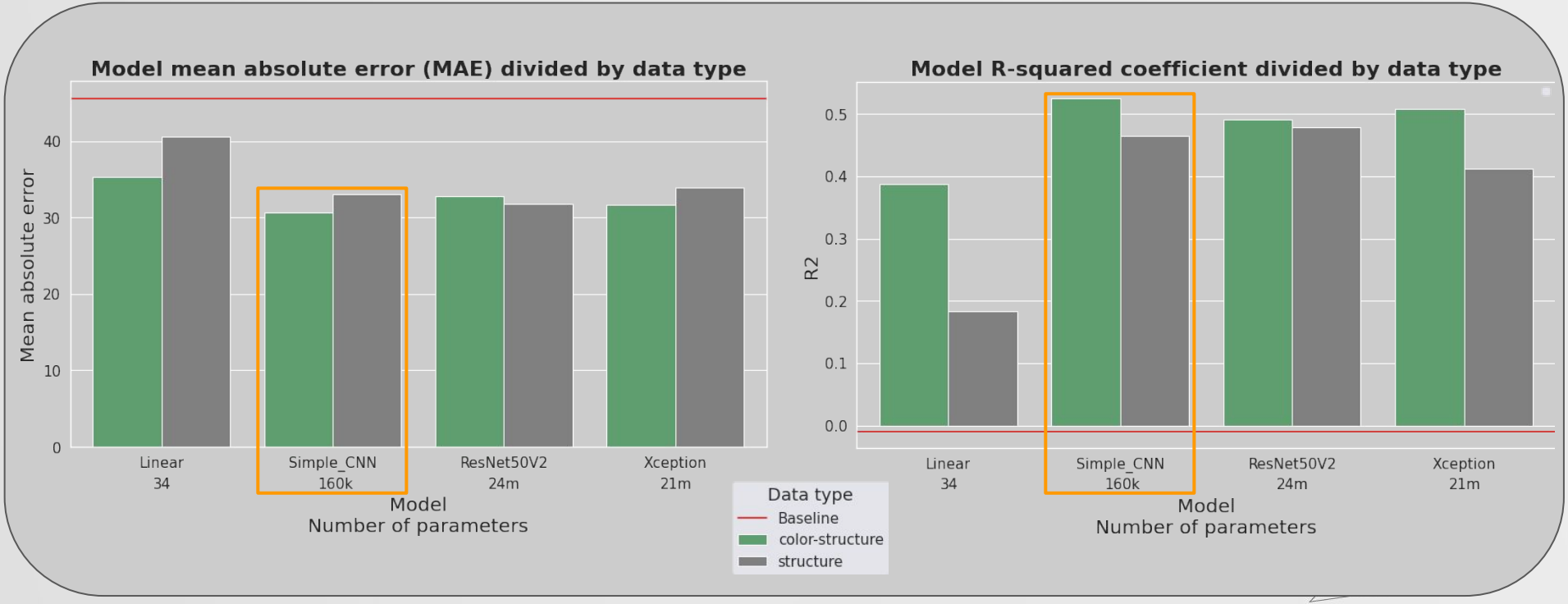


[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "ImageNet classification with deep convolutional neural networks".

03

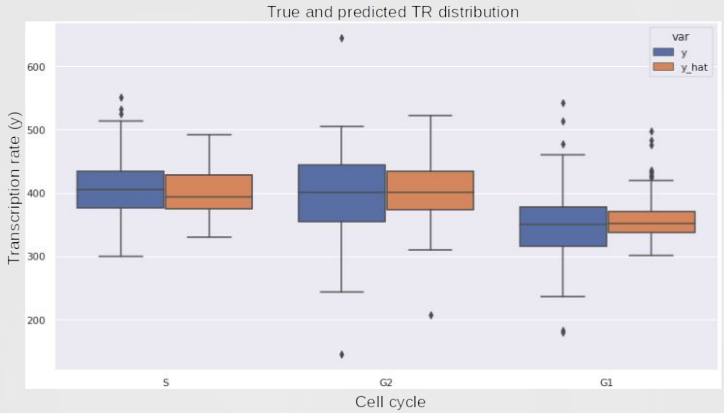
Model performance

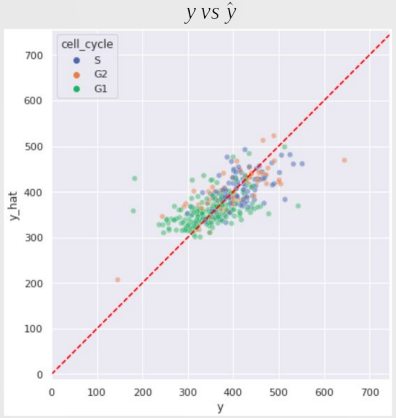
Model performance comparison



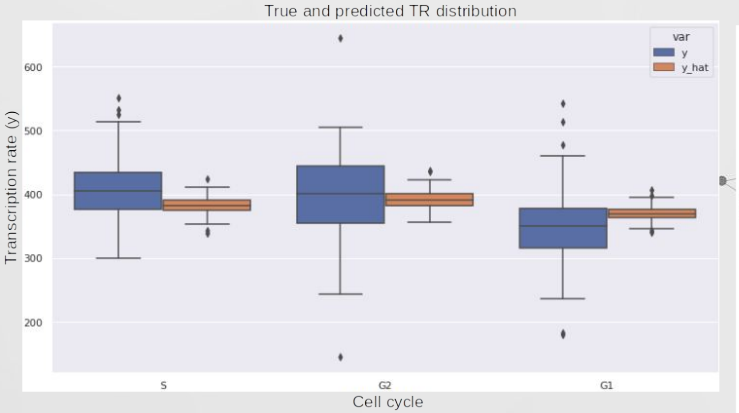
Model performance comparison

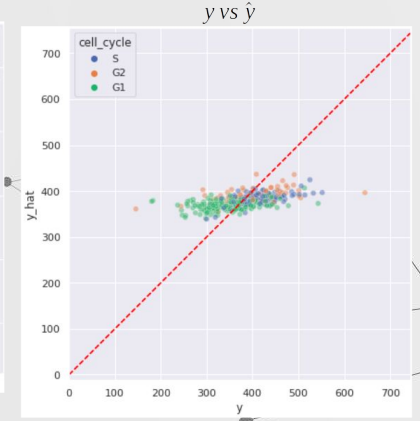
- **Simple CNN model** trained only with spatial information



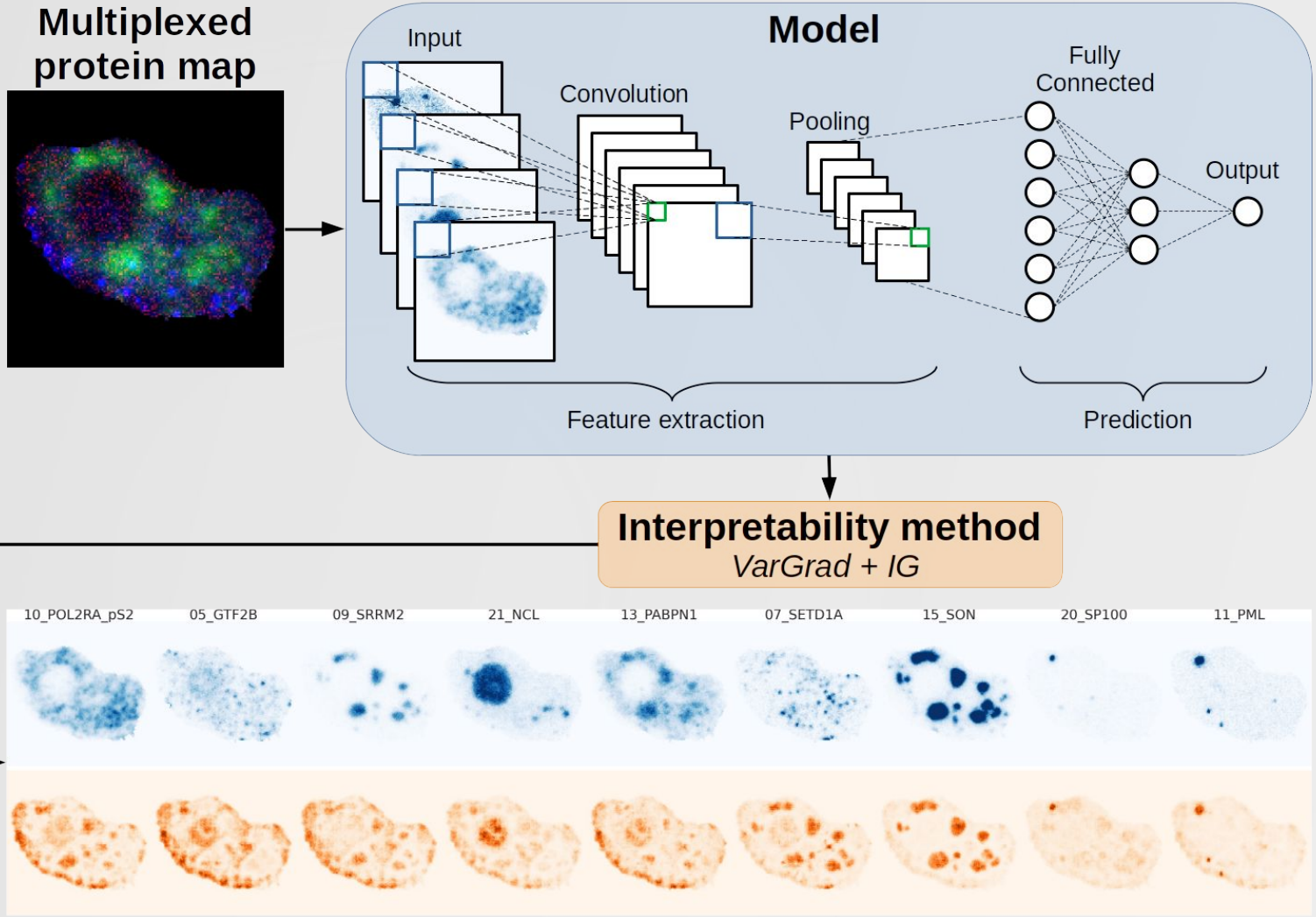


- **Linear model** trained only with spatial information





Road Map


Multiplexed protein map
Model
Input
Convolution
Pooling
Fully Connected
Output
Feature extraction
Prediction
Interpretability method
VarGrad + IG
10_POL2RA_pS2
05_GTF2B
09_SRRM2
21_NCL
13_PABPN1
07_SETD1A
15_SON
20_SP100
11_PML

04

Interpretability methods

Interpretability methods

- *Attribution methods* assign a score to each input feature based on how much it contribute to the output of the model [1]
- In g*radient-based methods* the **score** assignment is based on the **gradient** [2]

$$\phi(f, x) := \nabla_x f$$
$$= \frac{\partial f}{\partial x}$$

- However, as the model learns the relationship between inputs and outputs, the gradients can approximate to 0 (**saturation**) [3]

[1] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. Müller. “How to Explain Individual Classification Decisions”.

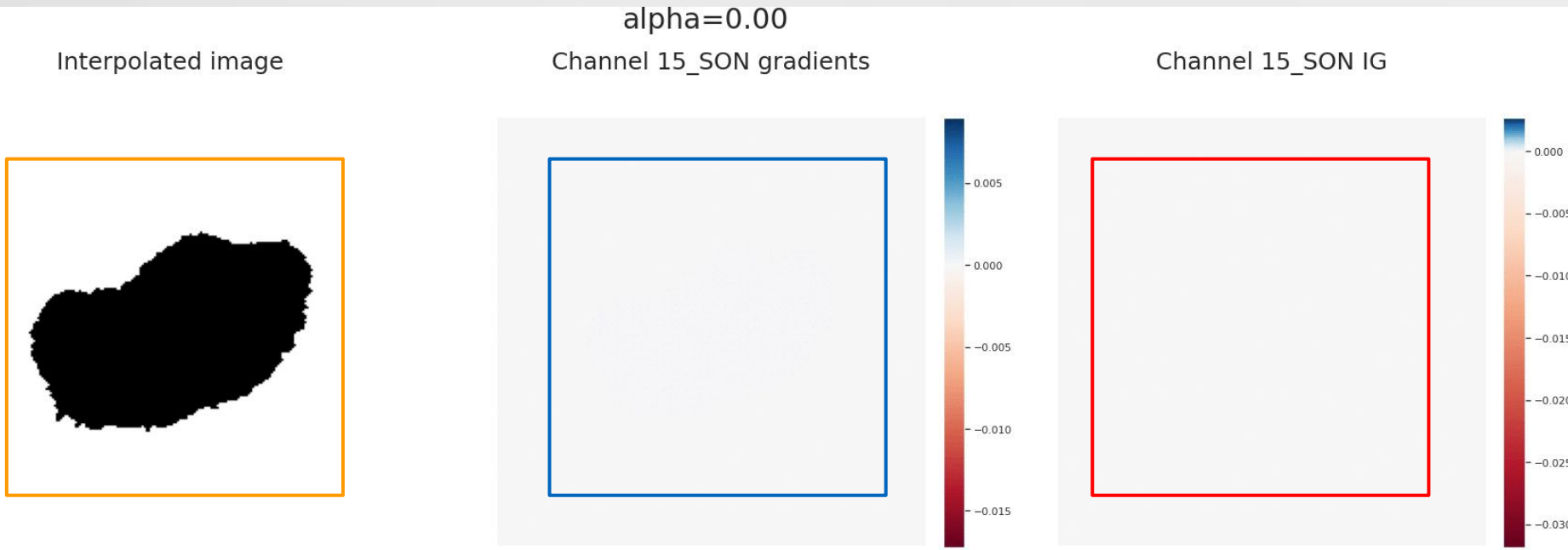
[2] P. Sturmfels, S. Lundberg, and S.-I. Lee. “Visualizing the Impact of Feature Attribution Baselines”.

[3] M. Sundararajan, A. Taly, and Q. Yan. “Axiomatic Attribution for Deep Networks”.

Interpretability methods: Integrated Gradients (IG)

To alleviate saturation, IG accumulates the gradient over a progression from a baseline to the input image [1]

$$\phi_i^{IG}(f, x, x') := (x_i - x_i') \int_{\alpha=0}^{1} \frac{\partial f(x' + \alpha(x - x'))}{\partial x_i} d\alpha$$



[1] M. Sundararajan, A. Taly, and Q. Yan. “Axiomatic Attribution for Deep Networks”.

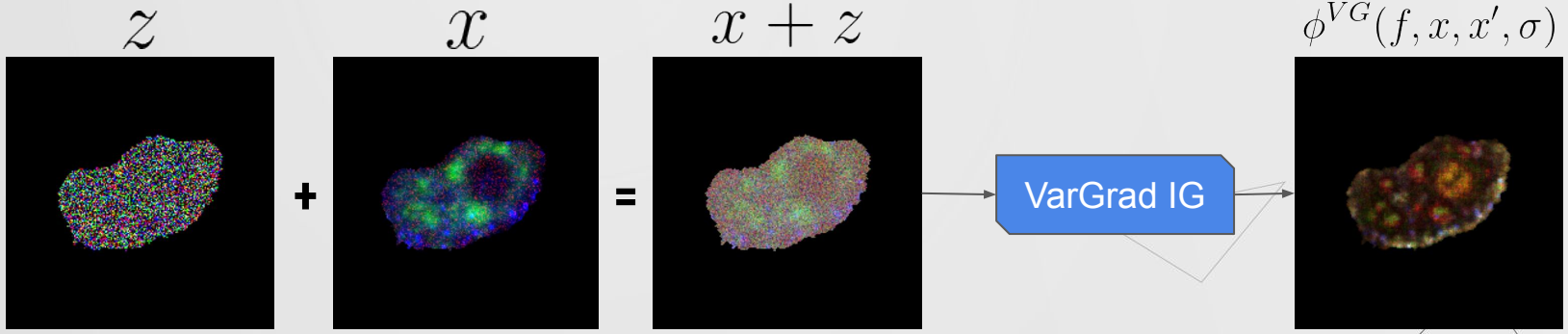
Interpretability methods: VarGrad IG

- However, attribution methods like IG are usually noisy and in some cases not better than a random designation of feature importance [1]
- Ensemble interpretability method like VarGrad [2] overcome simple methods like IG [3]

$$\phi^{VG}(f, x, x', \sigma) := Var(\Phi)$$

where $\Phi := \{\phi^{IG}(f, x + z, x') \mid z \in Z\}$, $Z := \{z_1, \ldots, z_M\} \subseteq \mathbb{R}^{D \times D \times C}$

and $z_{d_1,d_2,c} \sim \mathcal{N}(\mu, \sigma^2)$ are the i.i.d entrances of the noise image z



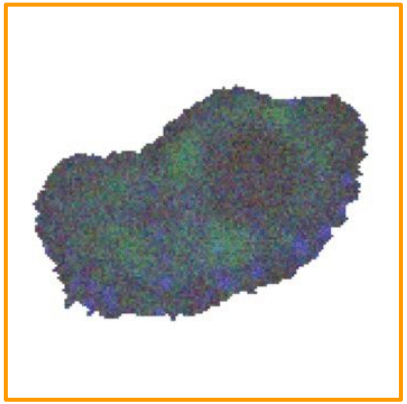
[1] S. Hooker, D. Erhan, P.-J. Kindermans, and B. Kim. "A benchmark for interpretability methods in deep neural networks".

[2] D. Smilkov, N. Thorat, B. Kim, F. B. Viégas, and M. Wattenberg. "Smooth-Grad: removing noise by adding noise".

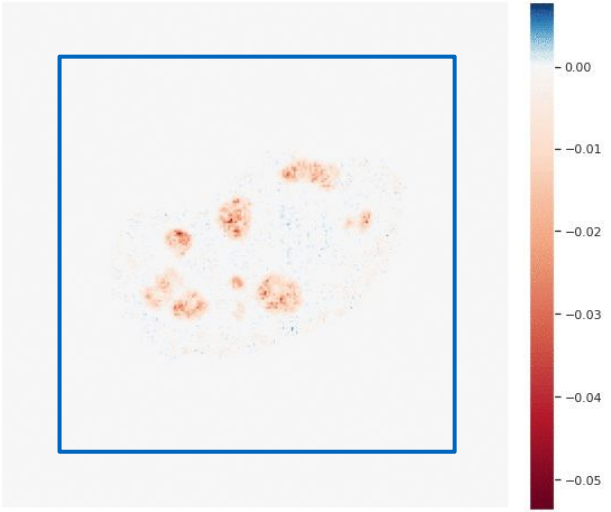
[3] J. Adebayo, J. Gilmer, I. Goodfellow, and B. Kim. "Local Explanation Methods for Deep Neural Networks Lack Sensitivity to Parameter Values".

N=1

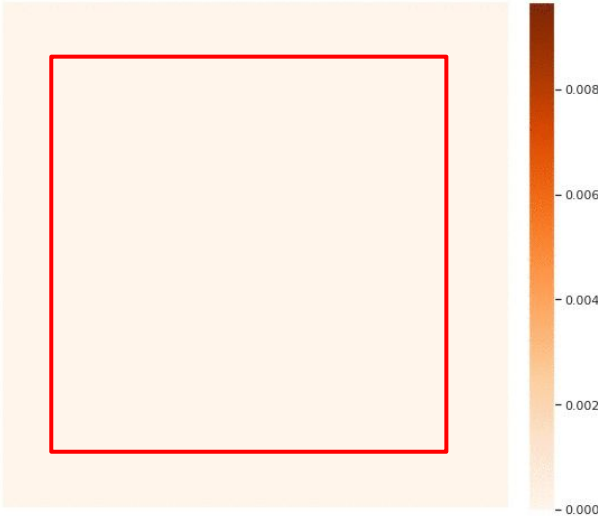
Interpolated image



Channel 15_SON IG



Channel 15_SON VG



$$\phi^{VG}(f, x, x', \sigma) := Var(\{\phi^{IG}(f, x + z, x) \mid z \in \{z_1, \ldots, z_M\} \subseteq \mathbb{R}^{D \times D \times C}\})$$

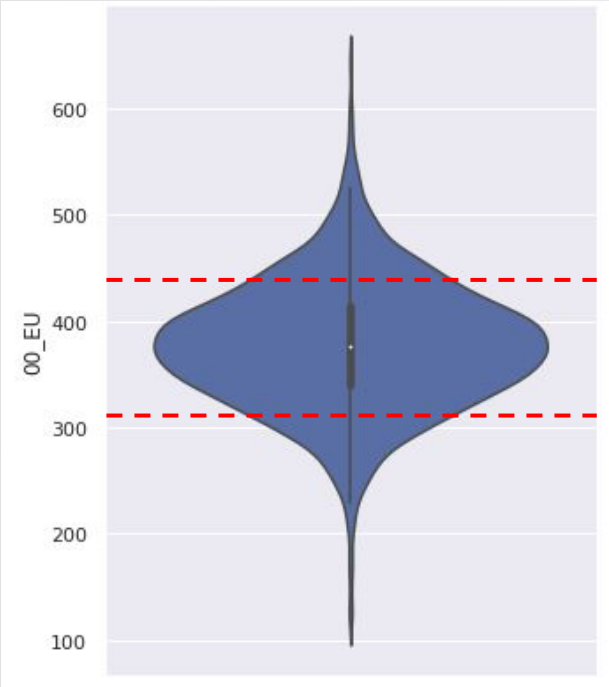
where $z_{d_1,d_2,c} \sim \mathcal{N}(\mu, \sigma^2)$, with $d_1, d_2 \in \{1, \ldots . D\}$ and $c \in \{1, \ldots . C\}$

05

Model interpretation

Experimental setup

- To classify the cells as low, medium and high TR, we use one standard deviation away from the mean TR
- Sample 3 cells with low, medium and high TR at random



High: if 438.02 <= TR

Medium: if 316.56 < TR < 438.02

Low: if TR <= 316.56

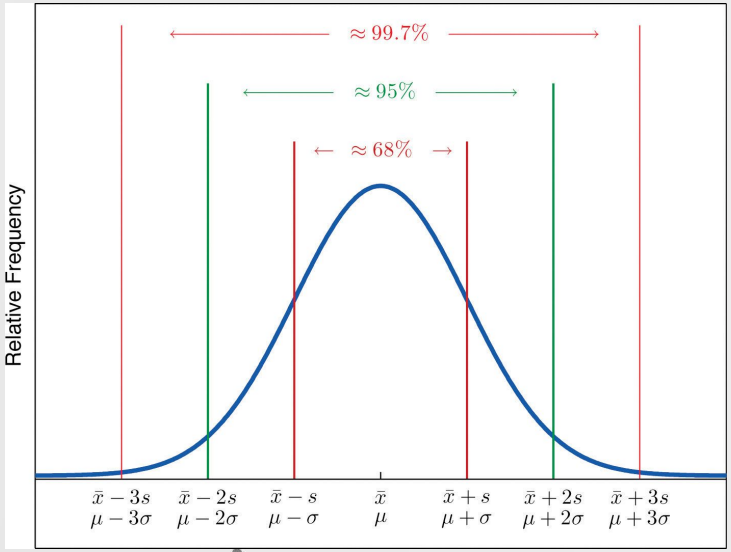
Or

High: 14.7% / 544 cells

Medium: 71% / 2627 cells

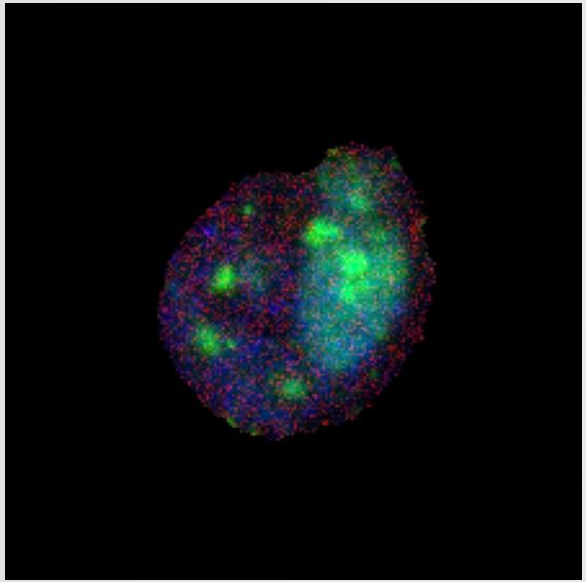
Low: 14.3 % / 532 cells

Chebyshev's Theorem



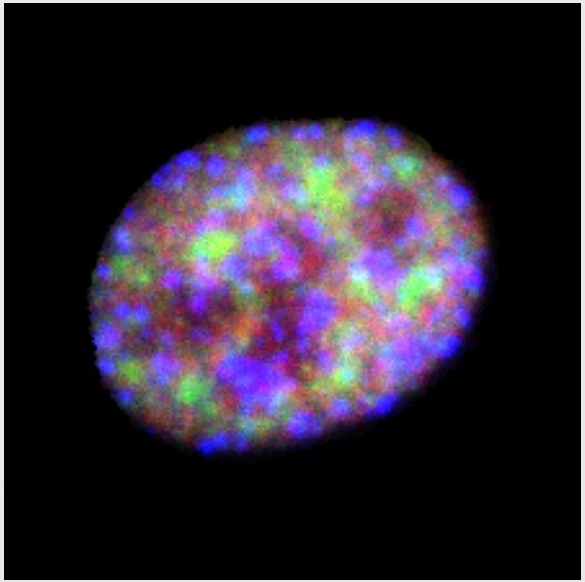
Experimental setup

Sample cell 1



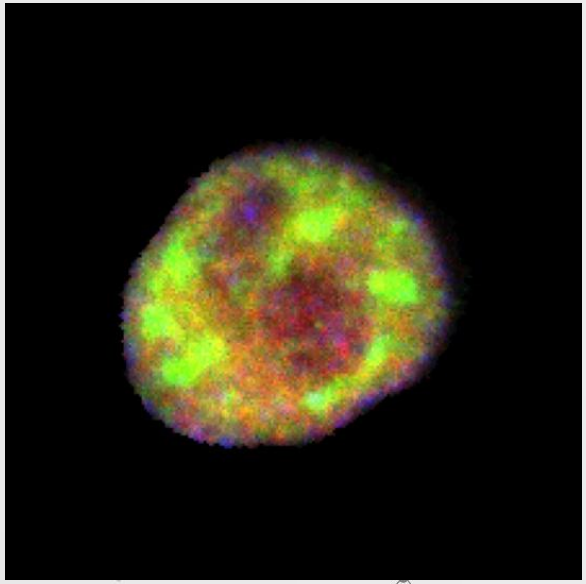
Cell id: 277417
TR: 133.04 (Low TR)
Cell Cycle: G1

Sample cell 2



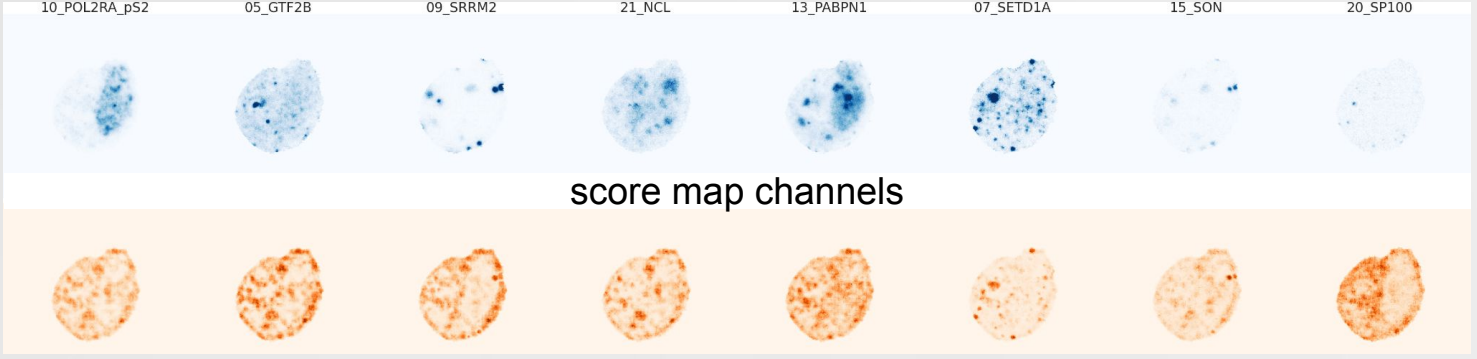
Cell id: 321001
TR: 378.19 (Medium TR)
Cell Cycle: S

Sample cell 3

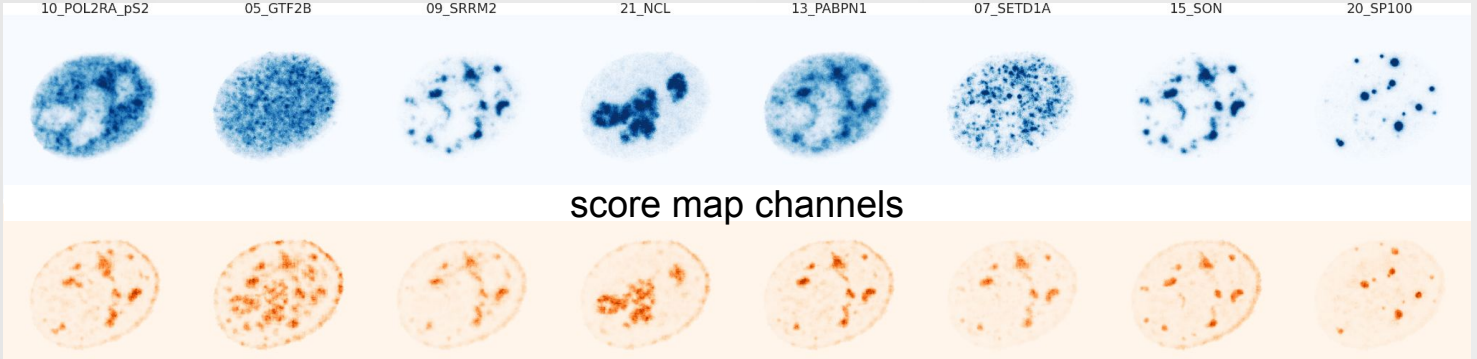


Cell id: 195536
TR: 540.09 (High TR)
Cell Cycle: G2

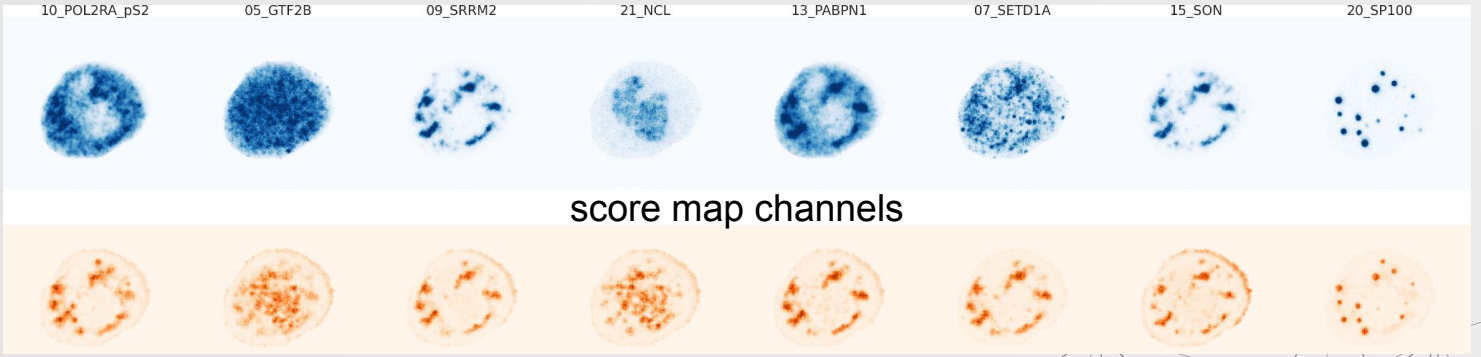
Cell id: 277417
TR: 133.04 (Low TR)
Cell Cycle: G1

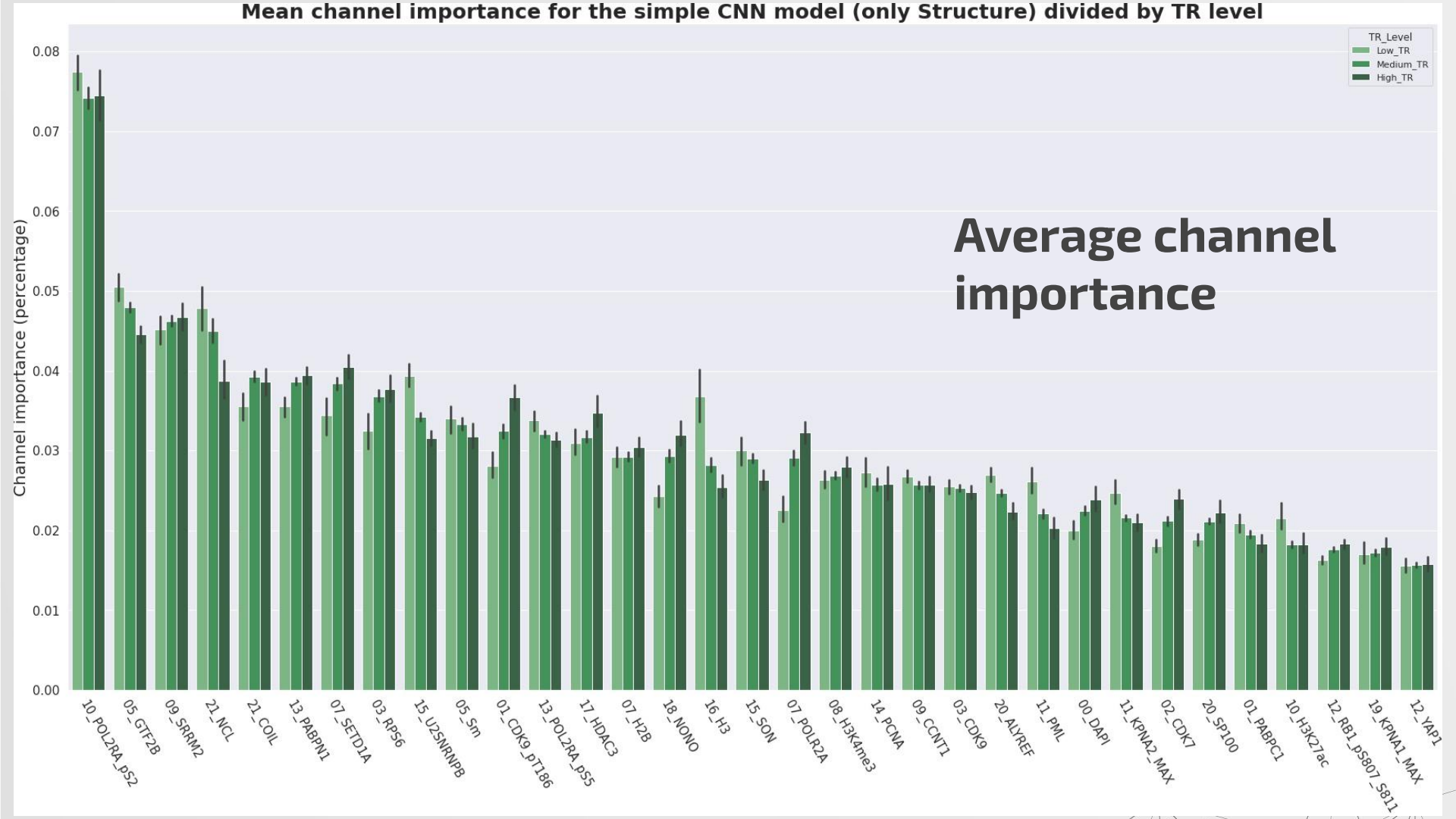


Cell id: 321001
TR: 378.19 (Medium TR)
Cell Cycle: S



Cell id: 195536
TR: 540.09 (High TR)
Cell Cycle: G2



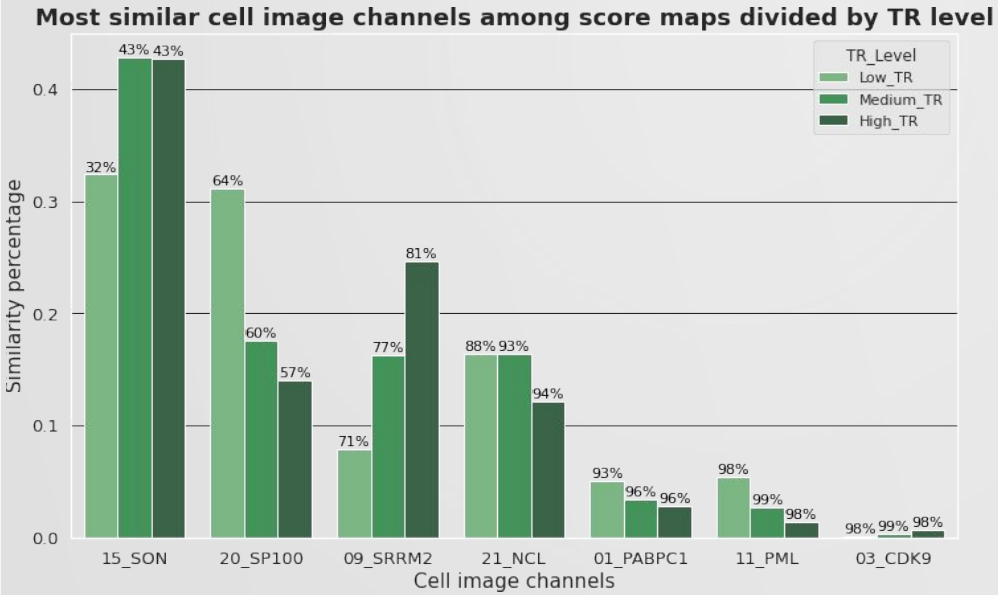

Mean channel importance for the simple CNN model (only Structure) divided by TR level
TR_Level
Low_TR
Medium_TR
High_TR
Channel importance (percentage)
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
Average channel importance
10_POL2RA_pS2
05_GTF2B
09_SRRM2
21_NCL
21_COIL
13_PABPN1
07_SETD1A
03_RPS6
15_U2SNRNPB
05_Sm
01_CDK9_pT186
13_POL2RA_pS5
17_HDAC3
07_H2B
18_NONO
16_H3
15_SON
07_POLR2A
08_H3K4me3
14_PCNA
09_CCNT1
03_CDK9
20_ALYREF
11_PML
00_DAPI
11_KPNA2_MAX
02_CDK7
20_SP100
01_PABPC1
10_H3K27ac
12_RB1_pS807_S811
19_KPNA1_MAX
12_YAP1

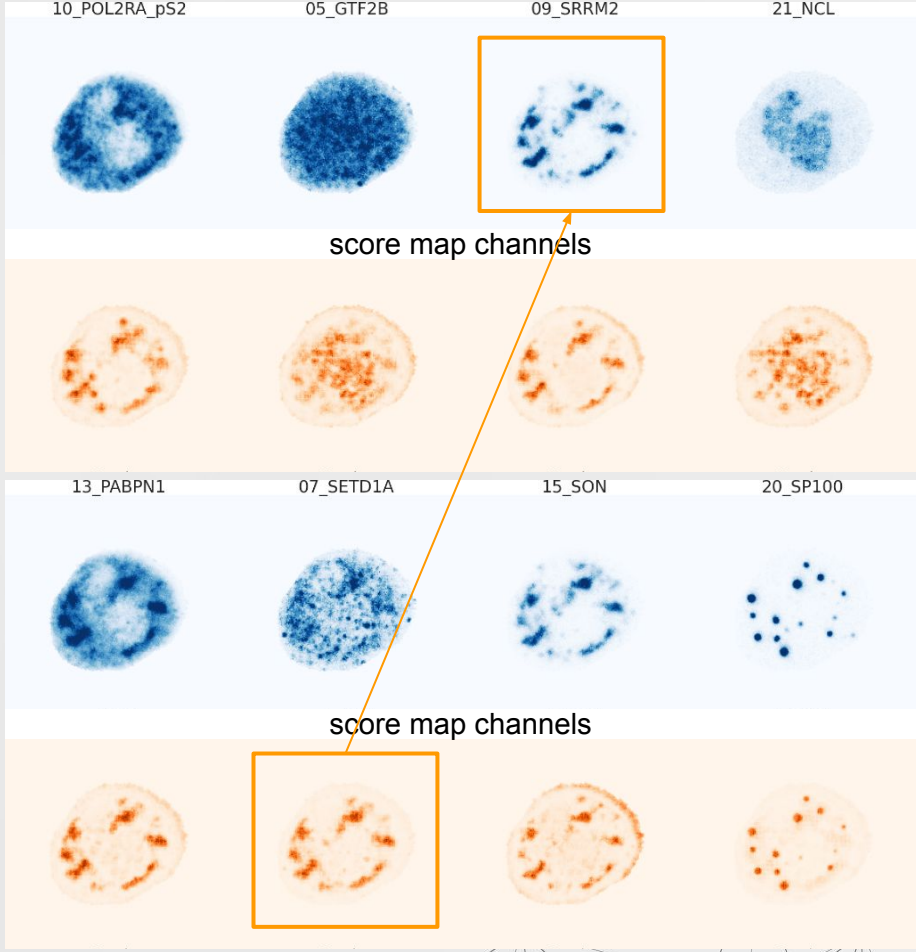
06

Score Maps similarity

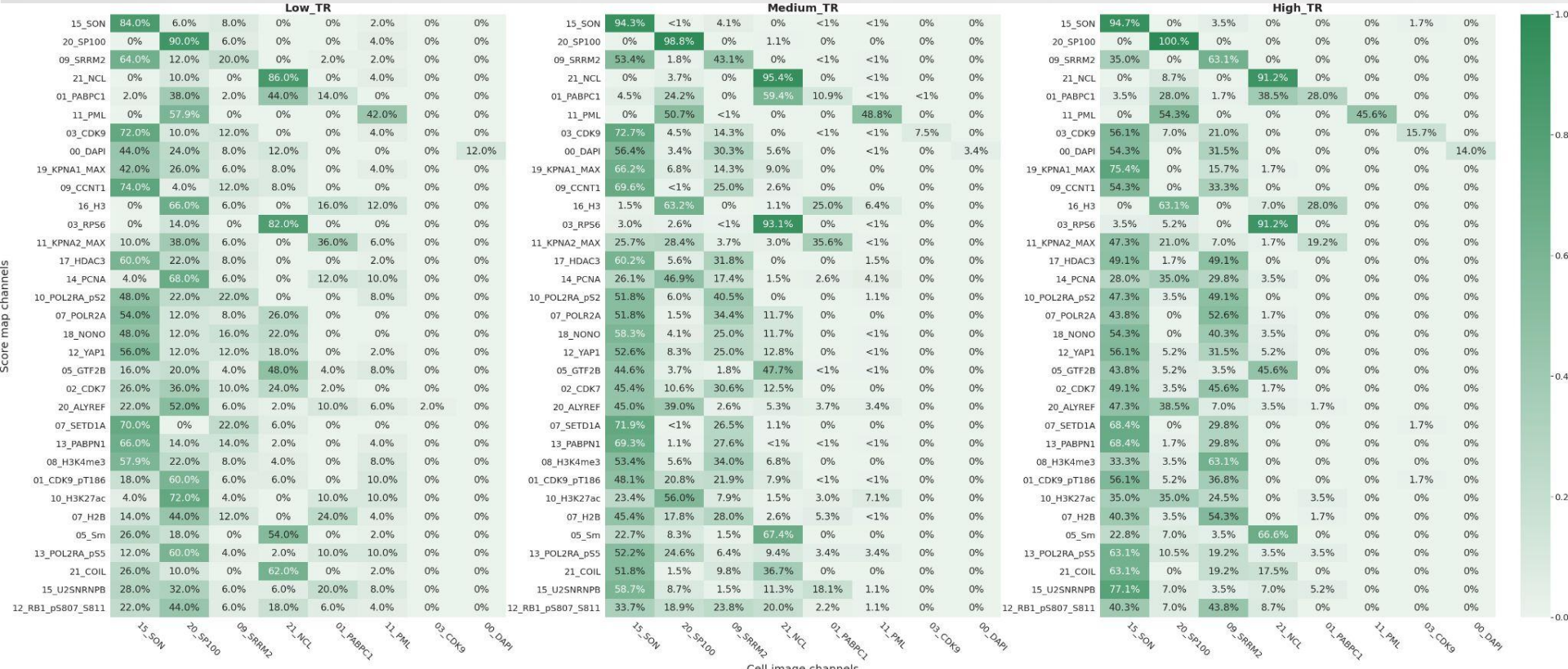
Similarity between Score Maps and Cell image

- Measure the similarity between score maps channels and cell image channels.
- Then, for each score map channel, select the cell image chanel most similar to it.

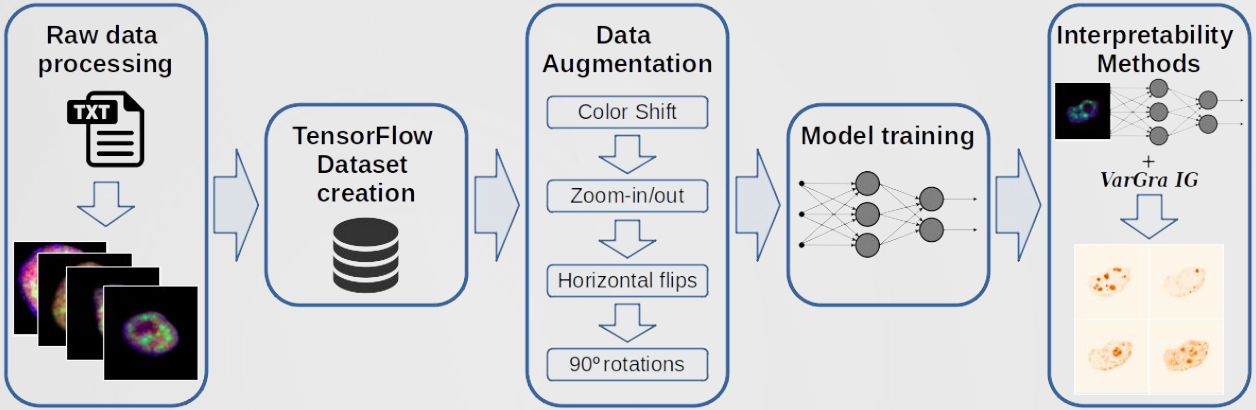




Most similar cell image channels to score map channels divided by transcription level



Outlook



- It is possible to predict TR using only localization information from the cell nucleus.
- Spatial organization of proteins inside the cell nucleus can be as important as the overall protein aboundas.
- It is important to generate data with subnuclear localization information, and develop methods able to use this data.
- Interpretability methods allow us know to which proteins and nuclear bodies were most relevant for the prediction of TR, and how this changes as the TR decreases or increases
- Interpretability methods allow us to learn from CNN models, which has the potential to provide guidance for new discoveries in the field of biology.

THANKS!

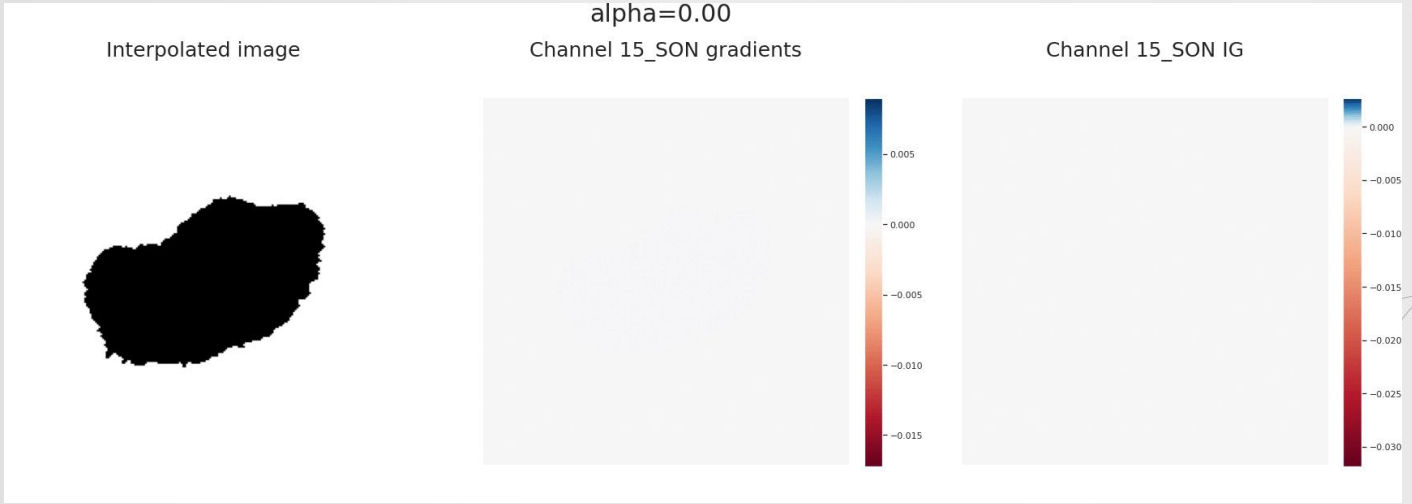
CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**.

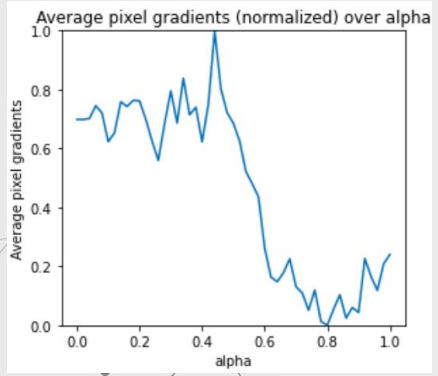
Please keep this slide for attribution.

Interpretability methods: Integrated Gradients (IG)

To alleviate saturation, IG accumulates the gradient over a progression from a baseline to the input image [1]

$$\phi_i^{IG}(f, x, x') := (x_i - x_i') \int_{\alpha=0}^{1} \frac{\partial f(x' + \alpha(x - x'))}{\partial x_i} d\alpha$$

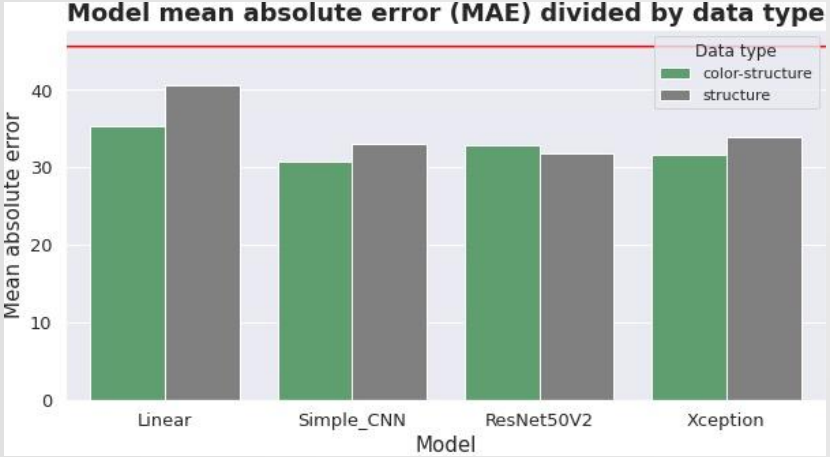


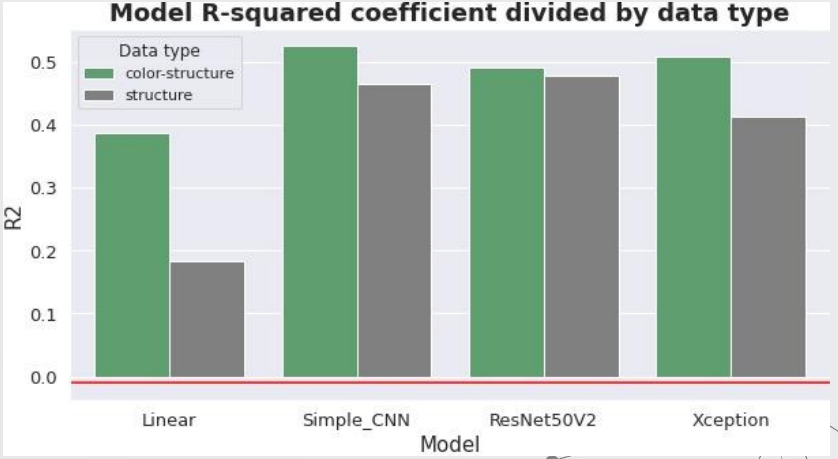


[1] M. Sundararajan, A. Taly, and Q. Yan. “Axiomatic Attribution for Deep Networks”.

Model performance comparison

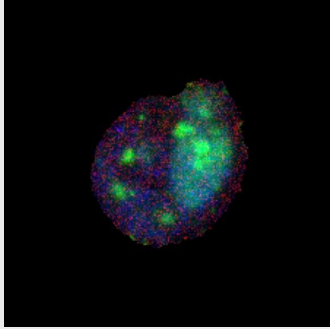
Model	Number of parameters	Data type	$\bar{e}$	$s(e)$	R^2	MAE	MSE	Huber
$\bar{y}$ (baseline)	0	targets avg	4.86	59.99	-0.01	45.56	3622	45.07
Linear	34	color-structure	4.06	46.83	0.38	35.26	2203	34.77
		structure	4.03	54.15	0.18	40.52	2941	40.02
Simple CNN	160k	color-structure	3.00	**41.27**	**0.52**	**30.68**	**1708**	**30.18**
		structure	0.77	43.94	0.46	33.08	1926	32.59
ResNet50V2	24m	color-structure	1.49	42.81	0.49	32.73	1830	32.24
		structure	**0.45**	43.38	0.47	31.83	1877	31.33
Xception	21m	color-structure	6.69	41.57	0.50	31.66	1768	31.16
		structure	7.23	45.50	0.41	33.92	2117	33.42

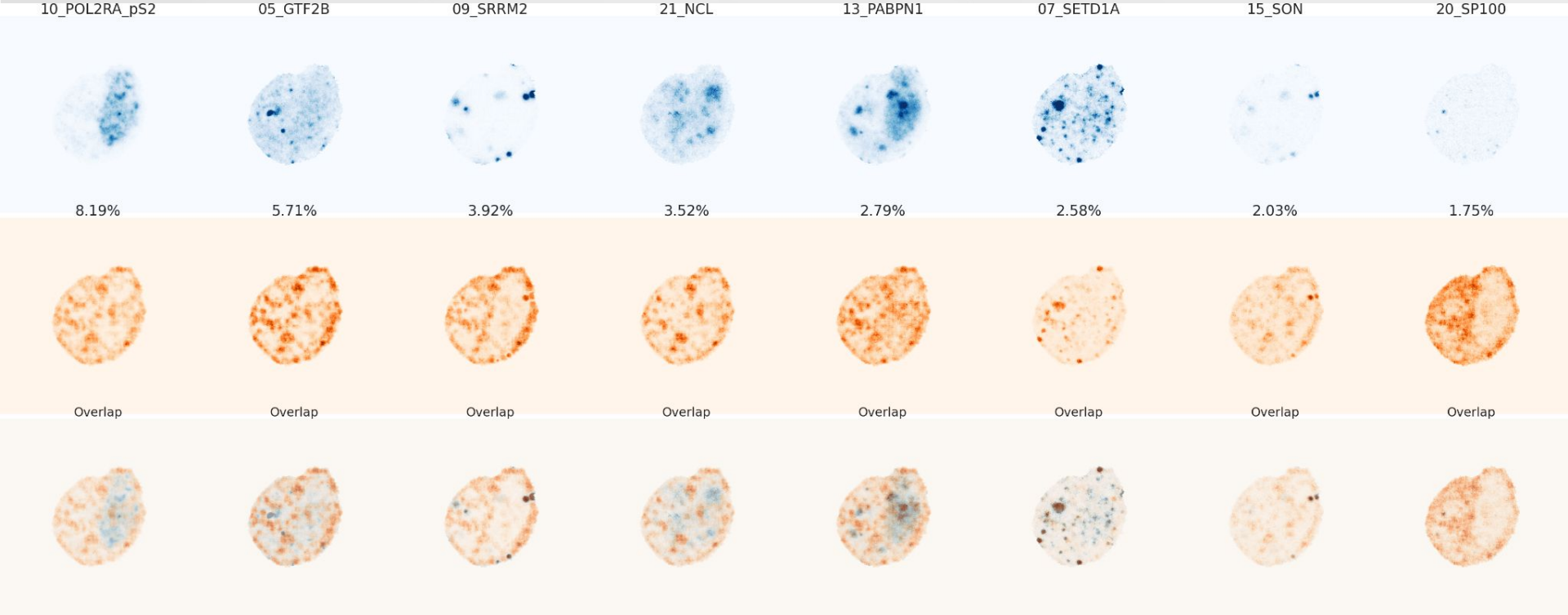




Most active channels during TR prediction for the simple CNN model

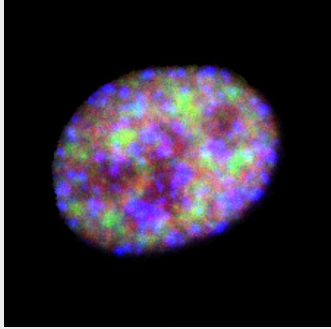
- Cell id: 277417
- TR: 133.04 (Low TR)
- Cell Cycle: G1

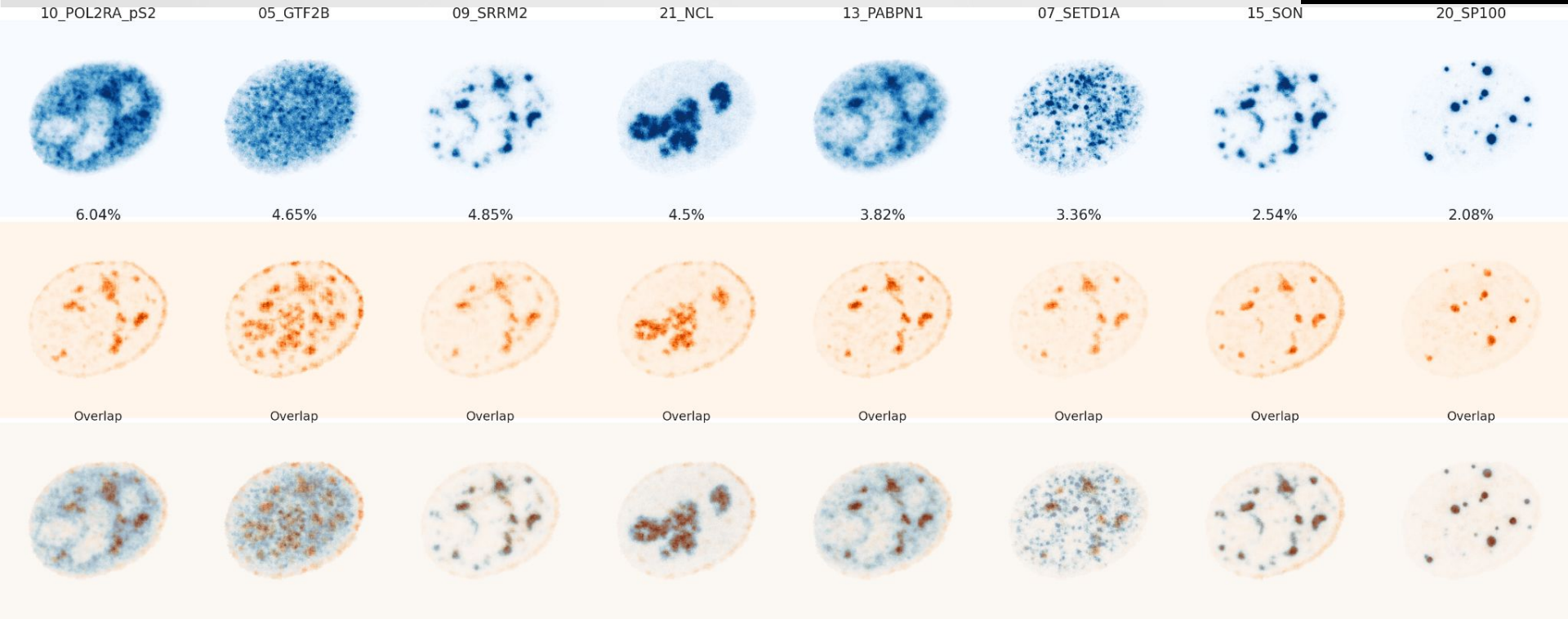




Most active channels during TR prediction for the simple CNN model

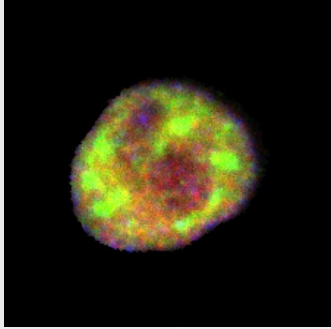
- Cell id: 321001
- TR: 378.19 (Medium TR)
- Cell Cycle: S

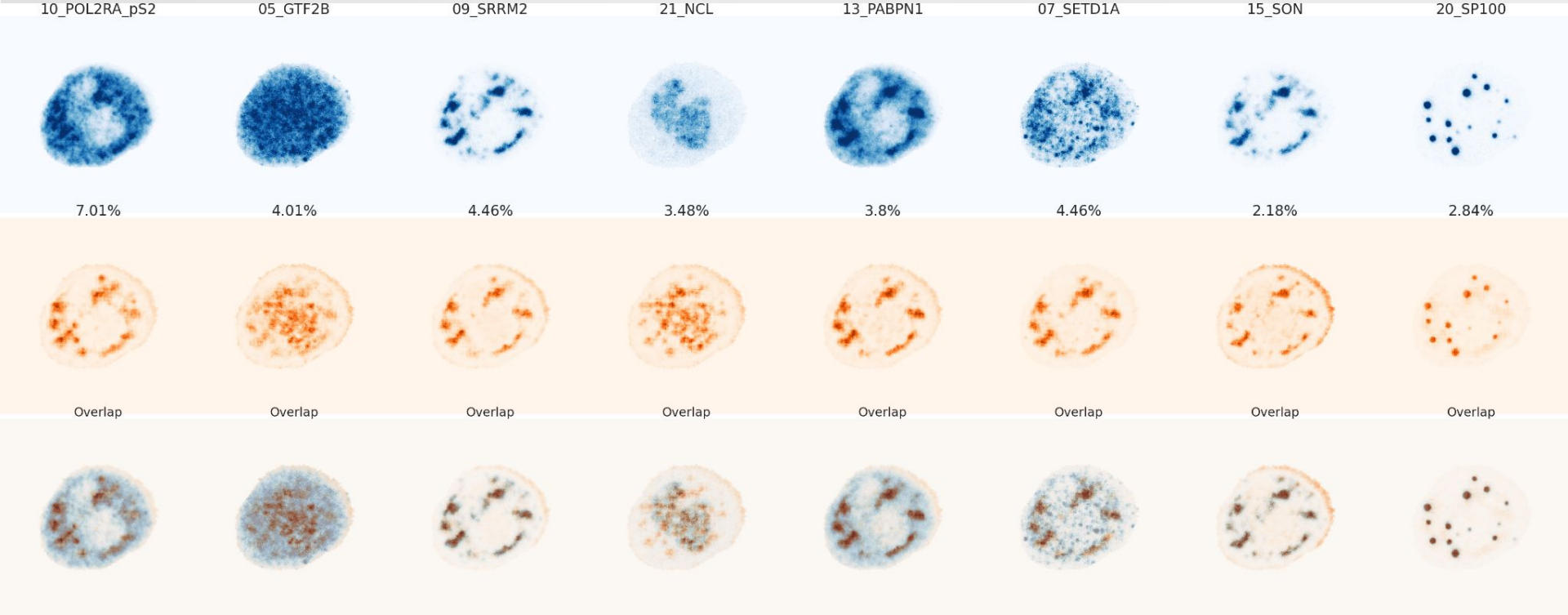




Most active channels during TR prediction for the simple CNN model

- Cell id: 195536
- TR: 540.09 (High TR)
- Cell Cycle: G2





ROAD MAP

01 Multiplexed protein maps

02 Data preprocessing and Data augmentation

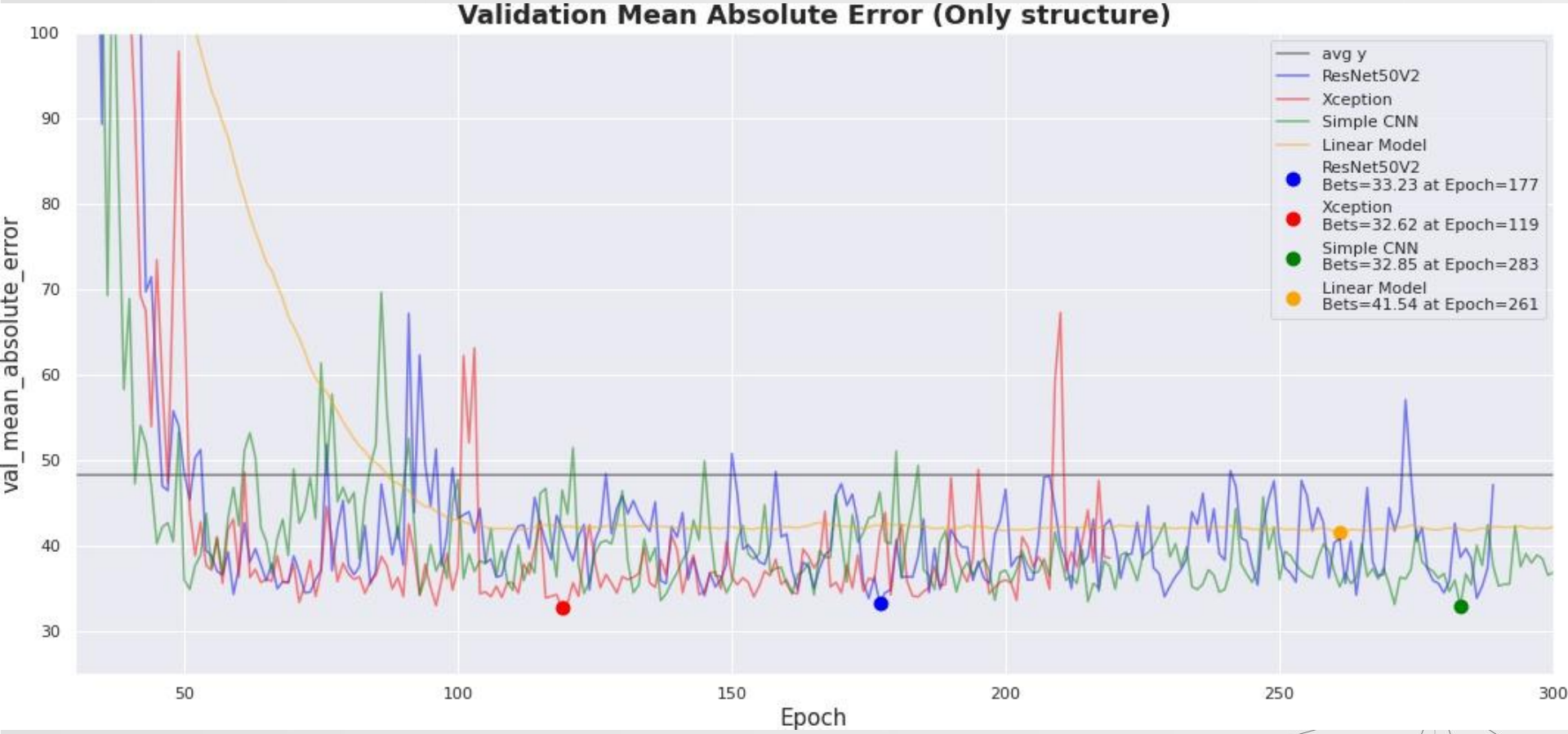
03 Model performance

04 Interpretability methods

05 Model interpretation

06 Score Maps similarity

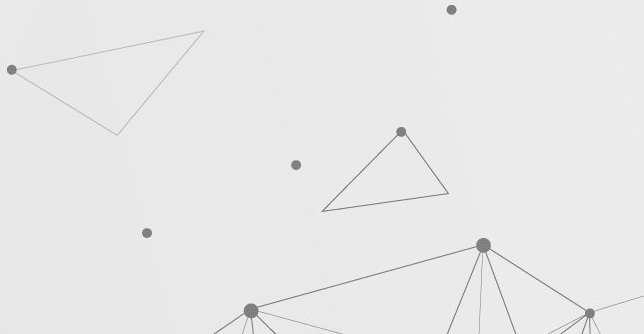
Model performance comparison



Cell cycle phase classification (G1, S, G2 and M)

Cell cycle phase was determined by means of a Support Vector Machine (SVM) classifier and k-means clustering.

1. A SVM classifier is trained to identify M phase cells based on the nuclear information.
2. A second SVM classifier is trained to identify cells in phase S.
3. Cells in phase G1 and G2 are classified using a k-means algorithm (excluding cells in S and M phase).

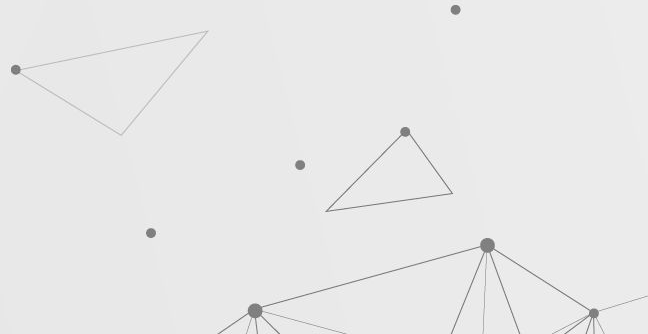


Linear model definition

- To see how much information is encoded only in the cell shape (and not in the color), we also used Linear models to predict the TR and compare the results with the CNN model results.
- Recall that a linear model is defined as:

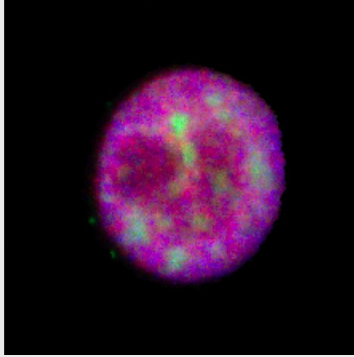
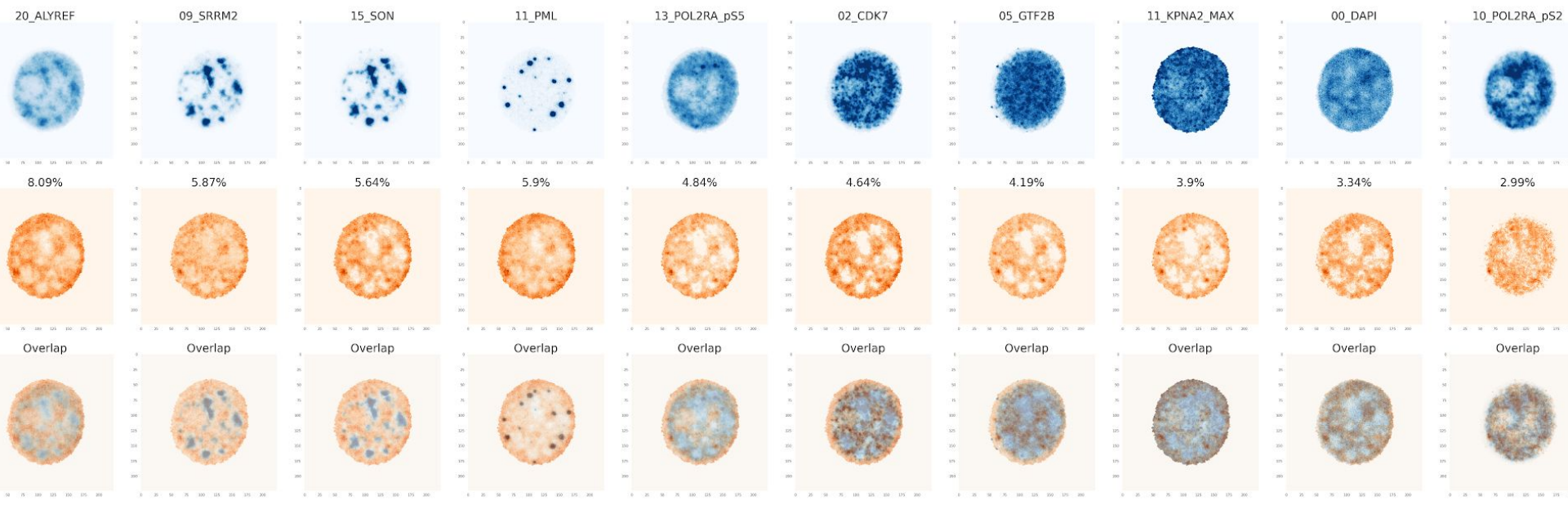
$$\hat{y}_i = \beta_0 + \beta_1 x_{i,1} + \cdots + \beta_d x_{i,d} + \alpha||\boldsymbol{\beta}||$$

where β_j (with $j \in \{1, \ldots, d\}$) are the coefficients for each of the input variables (in this case the average of each input channel), β_0 is the intercept, $\alpha||\boldsymbol{\beta}||$ the regularization term and $\hat{y}_i$ the target variable (in this case the average of the 00_EU channel, i.e. the TR).



Top 10 most active channels during TR prediction

without random color shifting



- Cell id: 262920
- TR: 513.5.17 (High TR)
- Cell Cycle: G2