Kernel-Based Multi-channel PolyCovNet

Team Borides

Kastan Day, Ruijie Zhu, Aria Coraor, Seonghwan Kim, Jiahui Yang

Contents

Predicting the lamellar period from monomer sequences and interaction parameters

A

B

C

Interaction Parameter

Lamellar Period
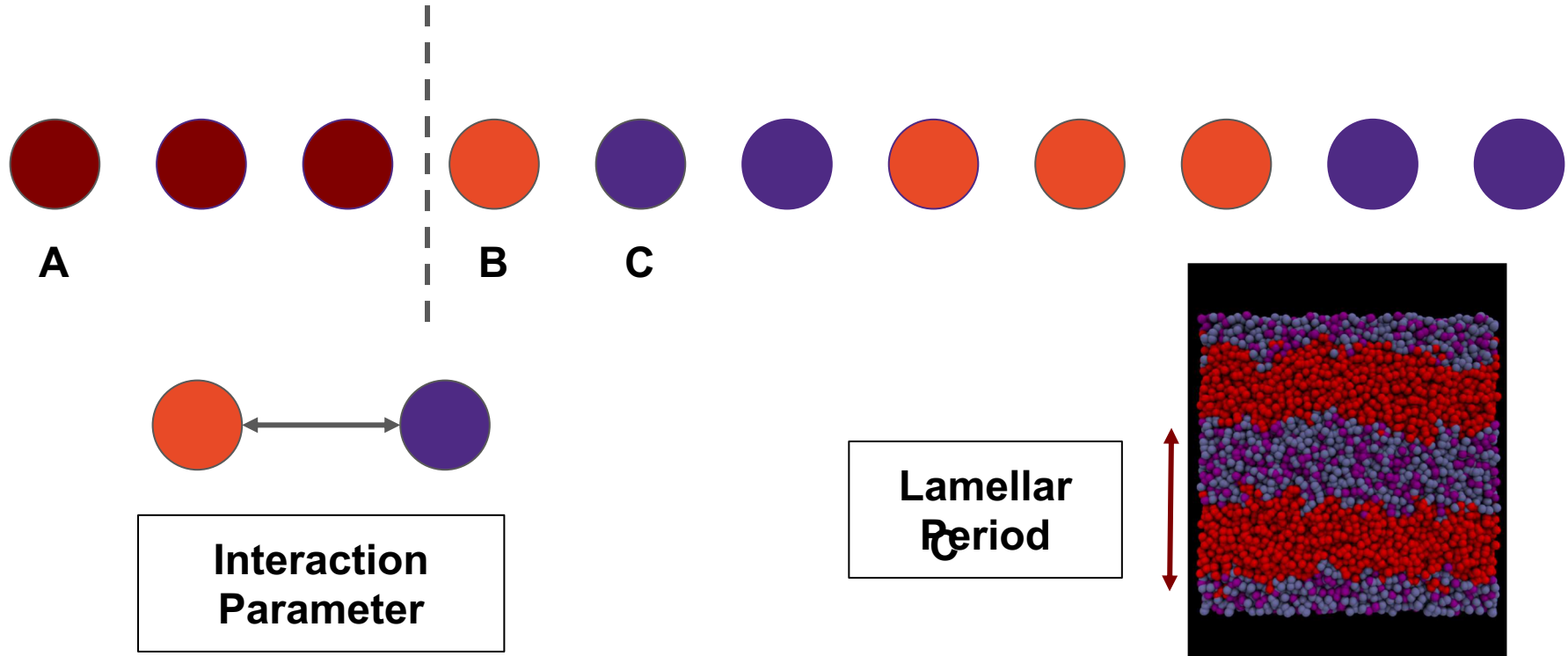
Sliding window - extract monomer sequence features

Sliding Window

window size

2

3

4

5

[0,0]
[0,1]
[1,0]
[1,1]

$2^2 = 4$

[0,0,0]
[0,0,1]
[0,0,0]
[0,0,0]
[0,0,1]
[0,0,0]
[0,0,0]
[0,0,1]

$2^3 = 8$

[0, 0, 0, 0]
[0, 0, 0, 1]
[0, 0, 1, 0]
[0, 0, 1, 1]
[0, 1, 0, 0]
[0, 1, 0, 1]
[0, 1, 1, 0]
[0, 1, 1, 1]
[1, 0, 0, 0]
[1, 0, 0, 1]
[1, 0, 1, 0]
[1, 0, 1, 1]
[1, 1, 0, 0]
[1, 1, 0, 1]
[1, 1, 1, 0]
[1, 1, 1, 1]

$2^4 = 16$

[0, 0, 0, 0, 0]
[0, 0, 0, 0, 1]
[0, 0, 0, 1, 0]
[0, 0, 0, 1, 1]
[0, 0, 1, 0, 0]
[0, 0, 1, 0, 1]
[0, 0, 1, 1, 0]
[0, 0, 1, 1, 1]
[0, 1, 0, 0, 0]
[0, 1, 0, 0, 1]
[0, 1, 0, 1, 0]
[0, 1, 0, 1, 1]
[0, 1, 1, 0, 0]
[0, 1, 1, 0, 1]
[0, 1, 1, 1, 0]
[0, 1, 1, 1, 1]
[1, 0, 0, 0, 0]
[1, 0, 0, 0, 1]
[1, 0, 0, 1, 0]
[1, 0, 0, 1, 1]
[1, 0, 1, 0, 0]
[1, 0, 1, 0, 1]
[1, 0, 1, 1, 0]
[1, 0, 1, 1, 1]
[1, 1, 0, 0, 0]
[1, 1, 0, 0, 1]
[1, 1, 0, 1, 0]
[1, 1, 0, 1, 1]
[1, 1, 1, 0, 0]
[1, 1, 1, 0, 1]
[1, 1, 1, 1, 0]
[1, 1, 1, 1, 1]

$2^5 = 32$
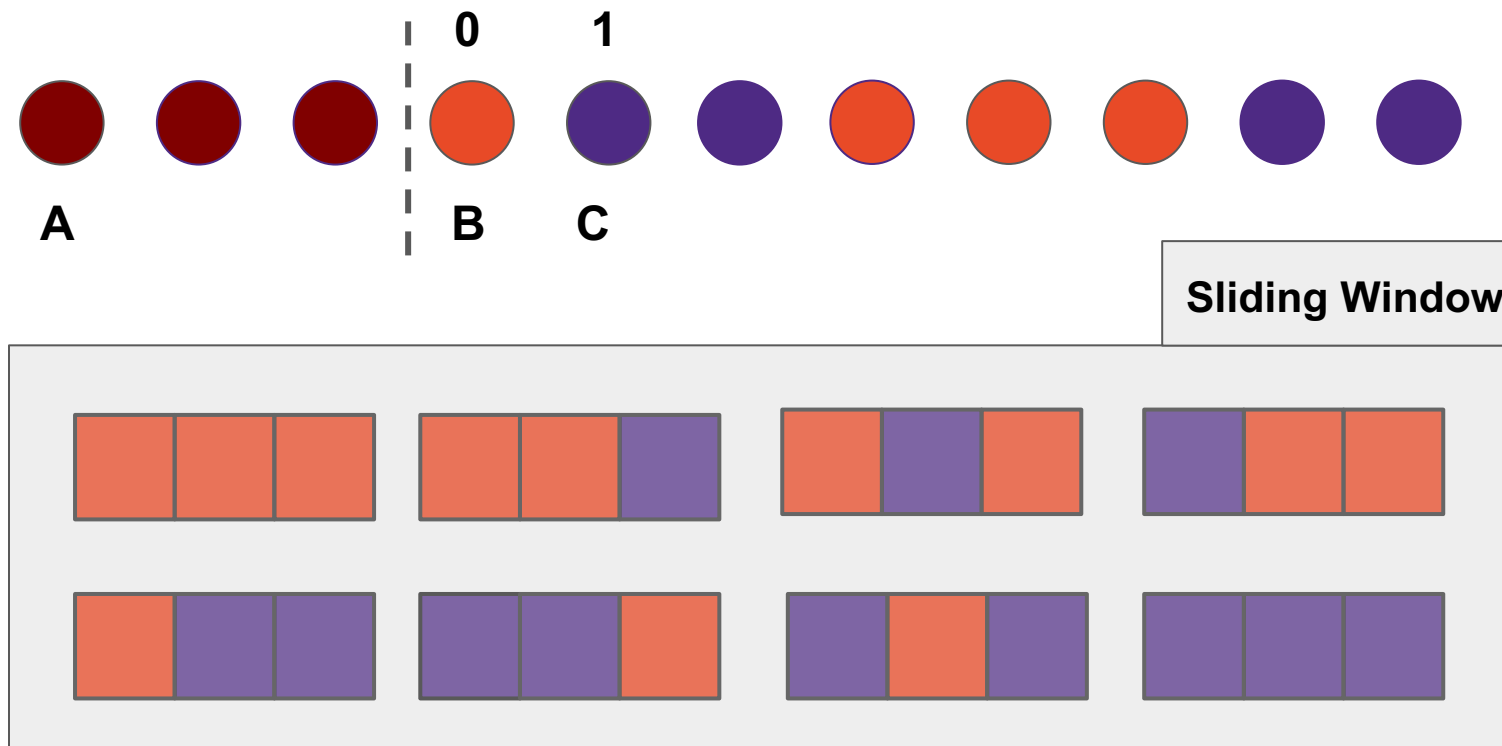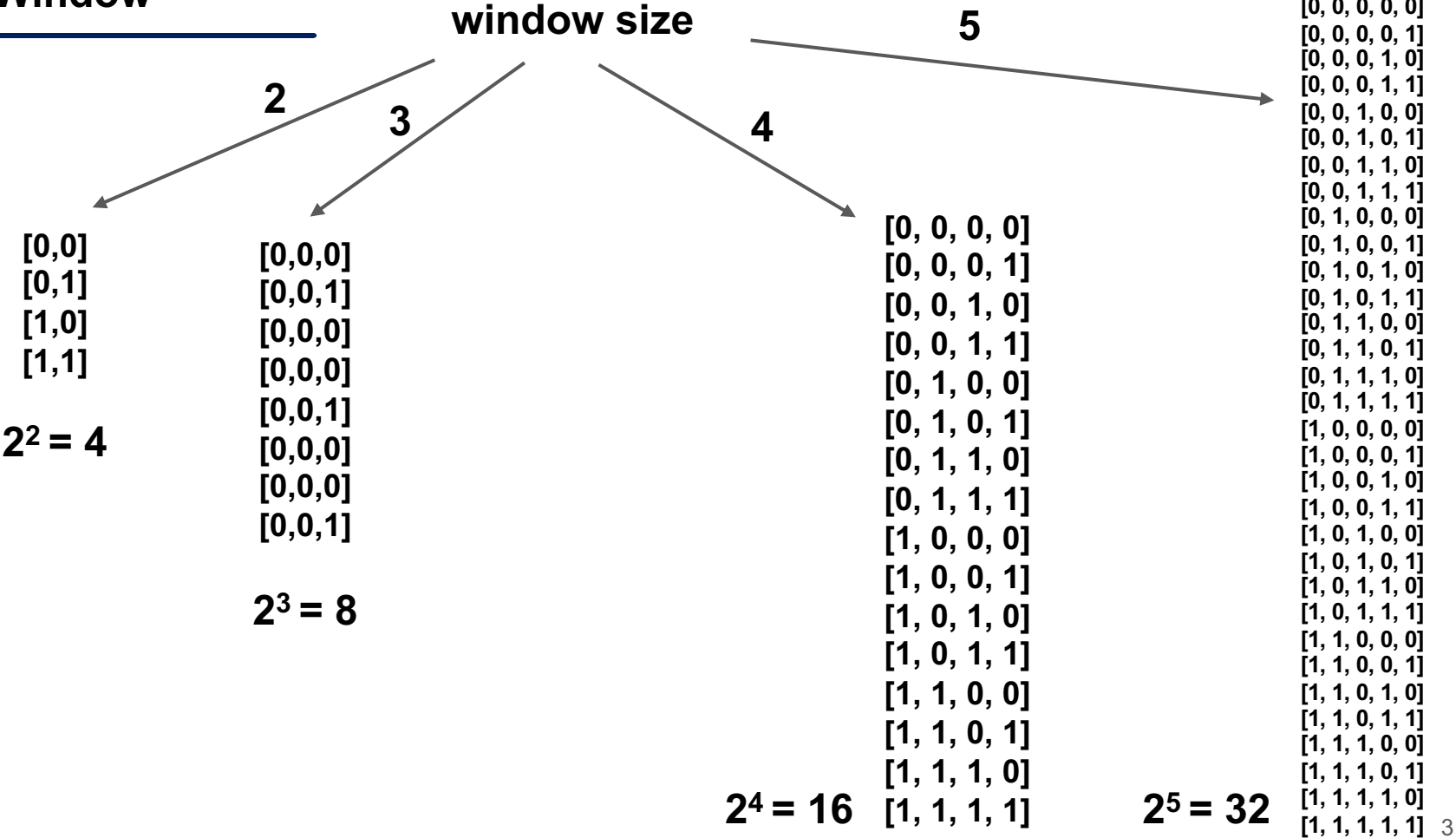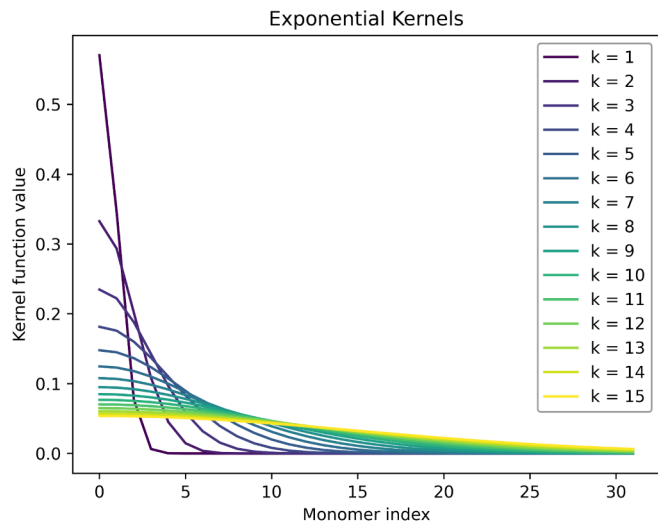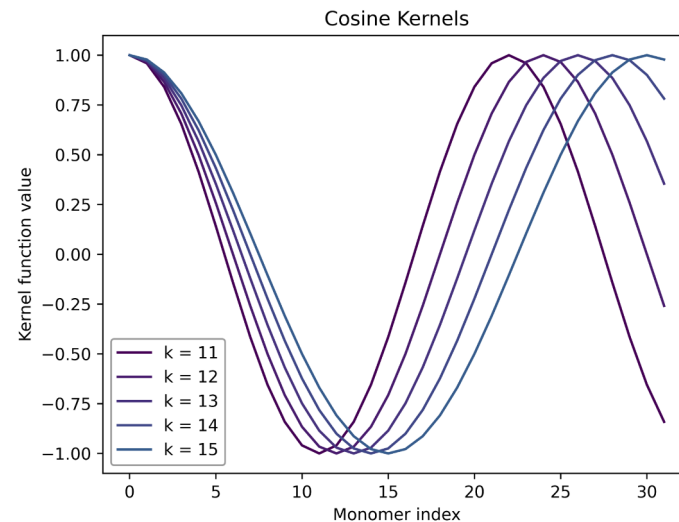
Non-linear kernel functions - preprocess monomer sequences

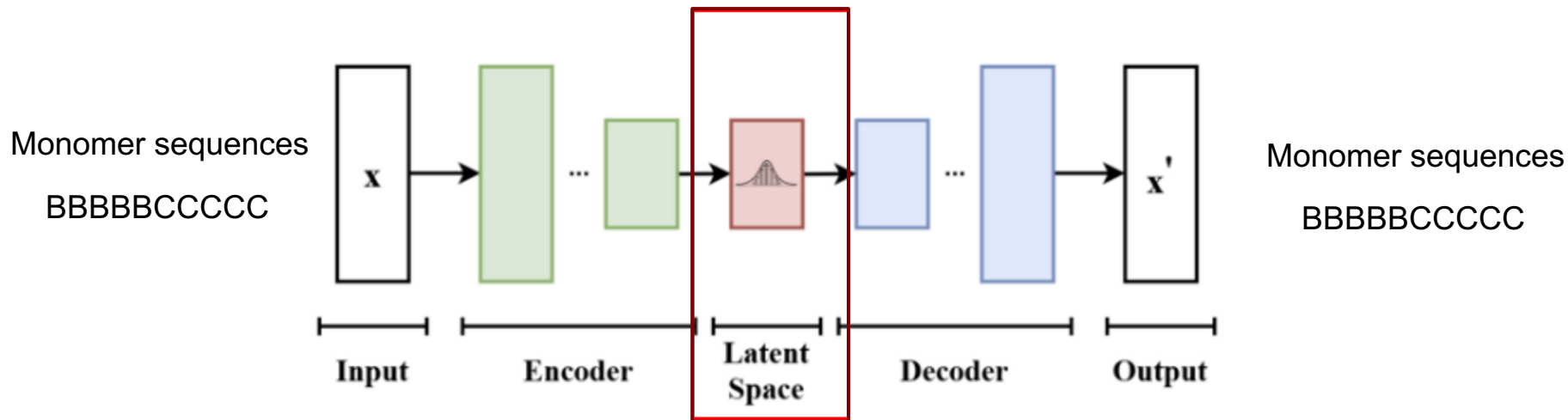1. Exponential kernels $\exp(\frac{x^2}{2k^2})$

2. Cosine kernels $\cos(\frac{\pi x}{k})$

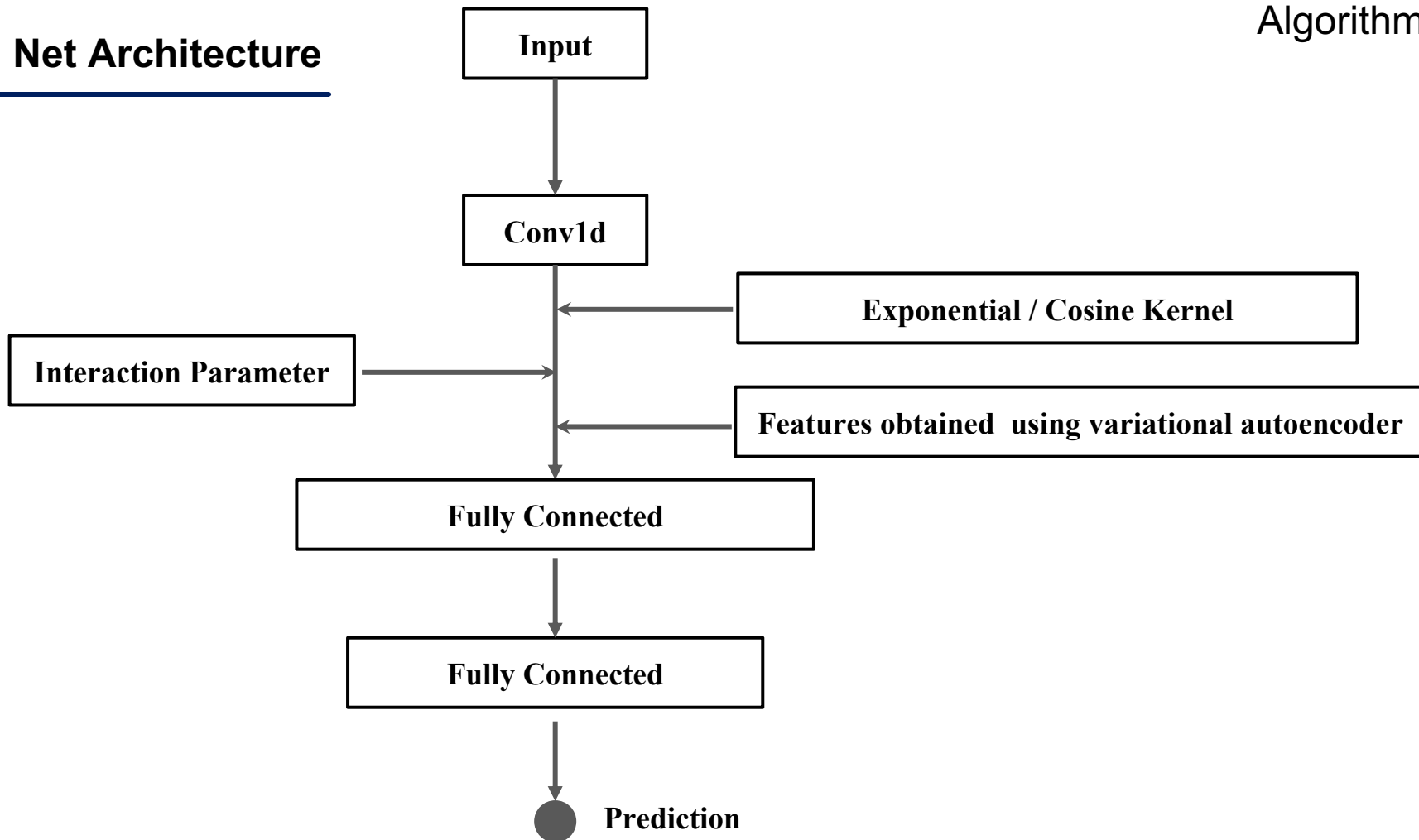Applying the above kernels on monomer sequences ➡ Non-linearity

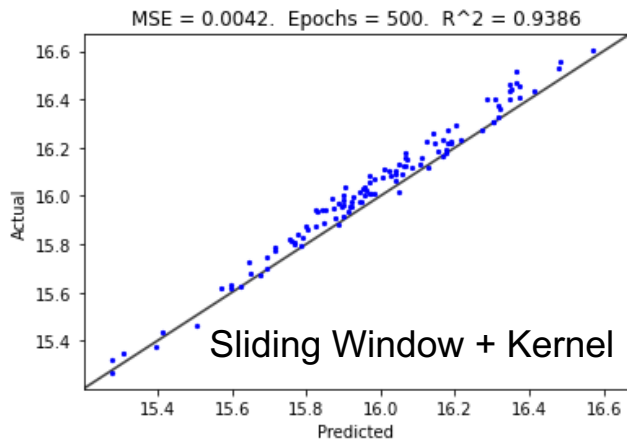# Variational autoencoder - extract features from monomer sequences

Monomer sequences

BBBBBCCCCC

Monomer sequences

BBBBBCCCCC

Reduced feature vectors

Nerual Net Architecture

w/o PC1

Results

MSE = 0.0042. Epochs = 500. R^2 = 0.9386

Sliding Window + Kernel

DataSetA

MSE = 0.0030. Epochs = 500. R^2 = 0.9645

Sliding Window + Kernel

DataSetB

MSE = 0.0814. Epochs = 500. R^2 = 0.8106

Sliding Window + Kernel

DataSetC

MSE = 0.0101. Epochs = 500. R^2 = 0.9496

Sliding Window + Kernel
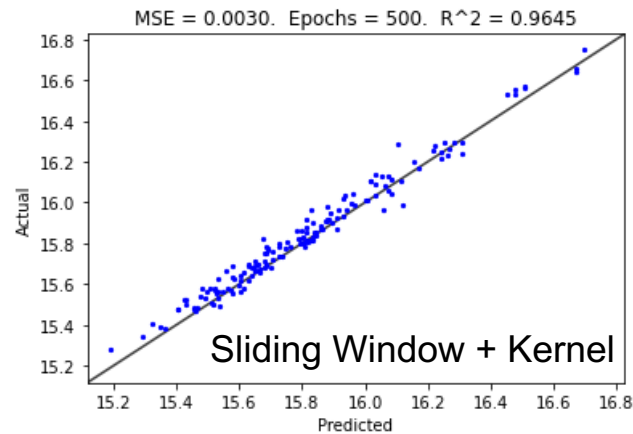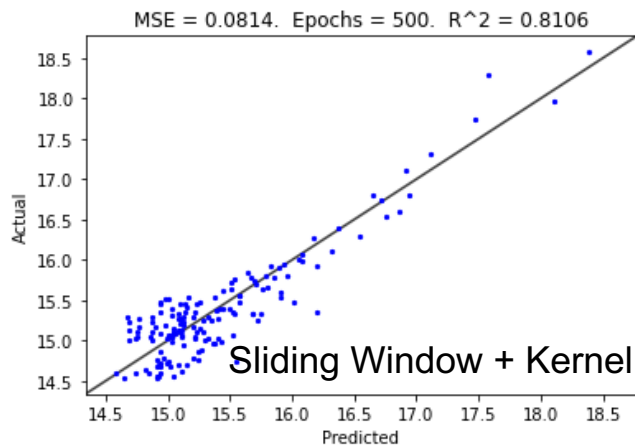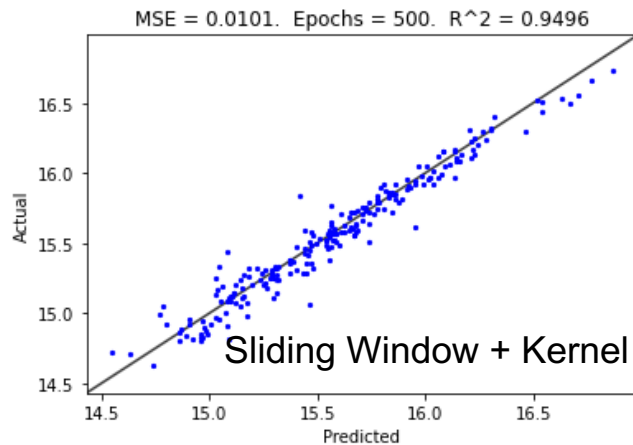
DataSetD

500 epoch
0.01 learning rate

7

w/PC 1

MSE = 0.0007. Epochs = 500. R^2 = 0.9942

DataSetA

Sliding Window+ Kernel+PC1

MSE = 0.0044. Epochs = 500. R^2 = 0.9839

DataSetB

Sliding Window+Kernel+PC1

MSE = 0.0539. Epochs = 500. R^2 = 0.8943

DataSetC

Sliding Window+Kernel+PC1-4

MSE = 0.0128. Epochs = 500. R^2 = 0.9627

DataSetD

Sliding Window+ Kernel+PC1
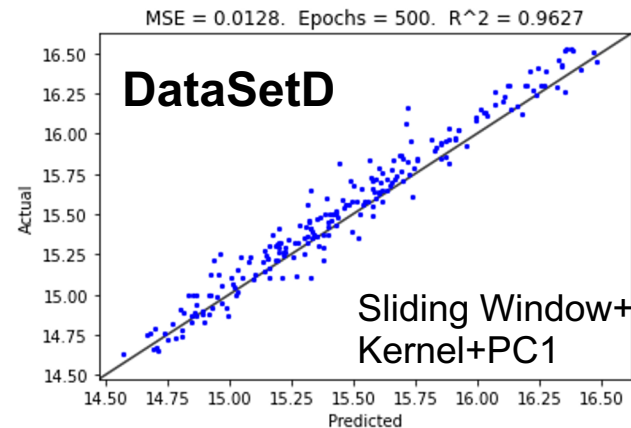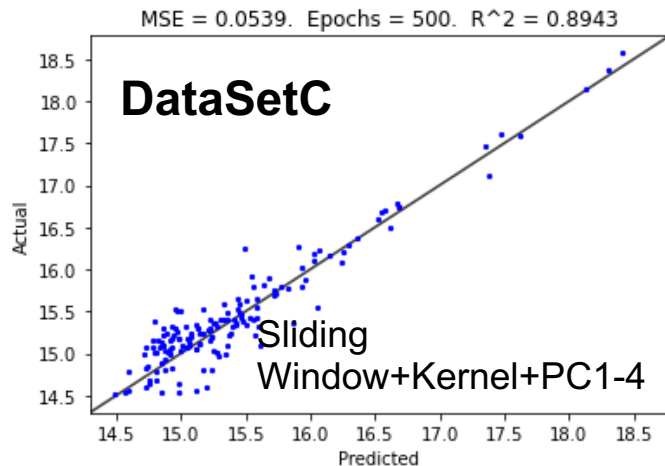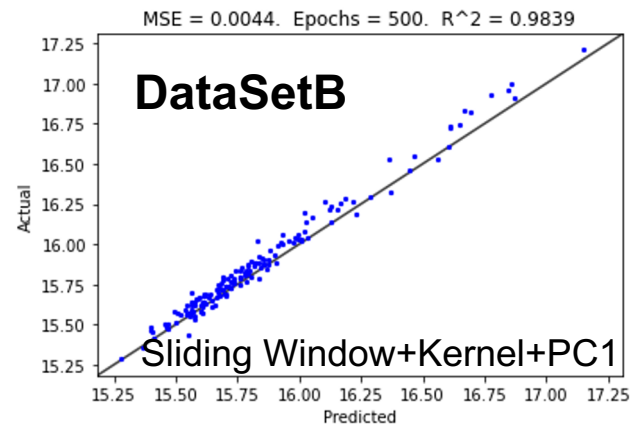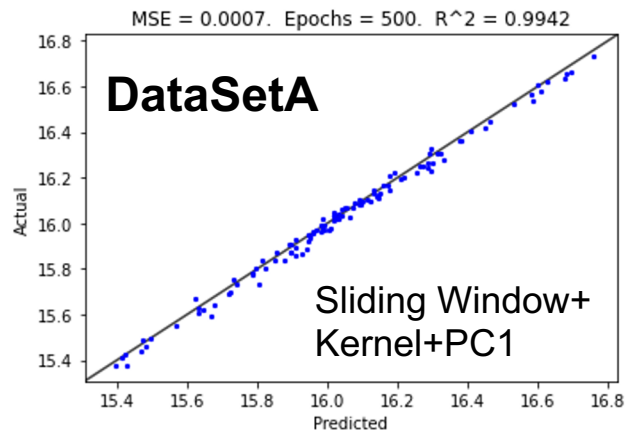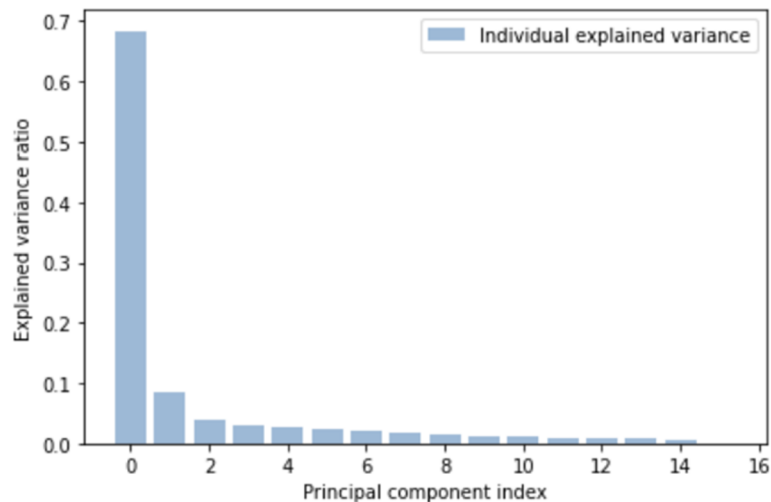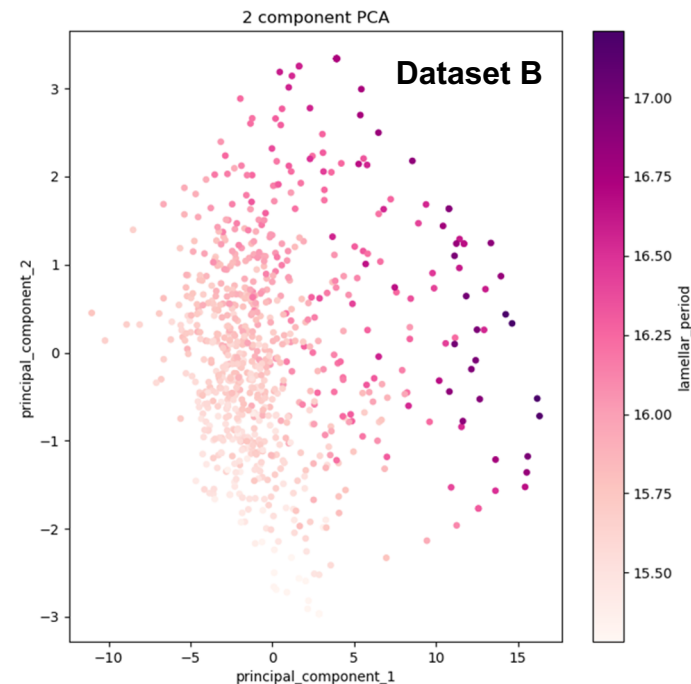
**500 epoch
0.01 learning rate**

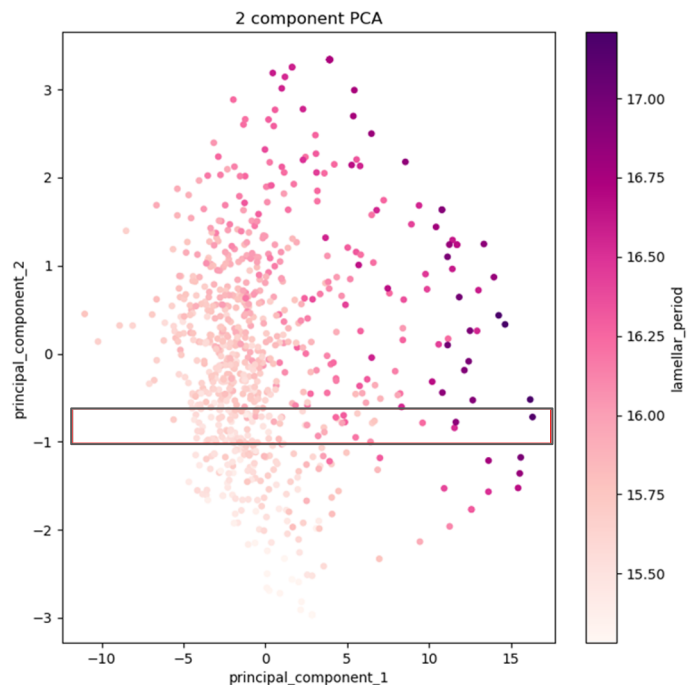## Principal component analysis on the latent space

Principal component 1 dominates

Larger principal component 1

Larger lamellar period

Blockiness is important to result in a high lamellar period

Higher principal component 1   **Blockiness**

111111111111111110000000000000000	0.875000
111111111111111110000000000000000	0.874510
111111111111111111000000000000000	0.873016
111111111111111111110000000000000	0.870445
111111111111111111111000000000000	0.866667

Lower principal component 1   **Blockiness**

110000110011100001100000000111111	0.498039
000000111111000000000000000111111	0.740891
110000011101101100000000001111110	0.372549
010001111101100100011010001000111	0.000000
101000011100010001001001000011011	-0.036437

Computational Efficiency (500 epochs)

Feature Generation	Time (min)
2-channel Sliding Window Features	0.5
Exponential / Cosine Kernel Features	0.08
VAE Features	30

Model Training / Validation	Time (min)
Training	1
Validation	0.02

* All runtimes are reported using ThetaGPU

Thank you!