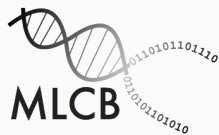# Presentation outline

Philip Hartout

April 29, 2022

MLCB

D BSSE

ETH *zürich*

Outline of presentations

1. Introduction
2. Library outline
3. How can graph kernels capture random noise in a protein point cloud? How does the graph representation of the protein influence MMD behaviour?
   - Vary ε and add gaussian noise to point cloud prior to graph extraction

Outline of presentations

1. Introduction
2. Library outline
3. How can graph kernels capture random noise in a protein point cloud? How does the graph representation of the protein influence MMD behaviour?
   - Vary ε and add gaussian noise to point cloud prior to graph extraction
4. How can we leverage kernels to capture geometric changes in a protein?
   - Topological features & kernels
   - Compare with W-L kernel
   - Compare also with Ramachandran descriptors

Outline of presentations

1. Introduction
2. Library outline
3. How can graph kernels capture random noise in a protein point cloud? How does the graph representation of the protein influence MMD behaviour?
   - Vary ε and add gaussian noise to point cloud prior to graph extraction
4. How can we leverage kernels to capture geometric changes in a protein?
   - Topological features & kernels
   - Compare with W-L kernel
   - Compare also with Ramachandran descriptors
5. How can we capture sequence changes in a kernel?
   - ESM
   - W-L kernel
   - Spectrum kernel? Essentially very close to the W-L kernel.

Outline of presentations

1. Introduction
2. Library outline
3. How can graph kernels capture random noise in a protein point cloud? How does the graph representation of the protein influence MMD behaviour?
   - Vary ε and add gaussian noise to point cloud prior to graph extraction
4. How can we leverage kernels to capture geometric changes in a protein?
   - Topological features & kernels
   - Compare with W-L kernel
   - Compare also with Ramachandran descriptors
5. How can we capture sequence changes in a kernel?
   - ESM
   - W-L kernel
   - Spectrum kernel? Essentially very close to the W-L kernel.
6. W-L speed improvement