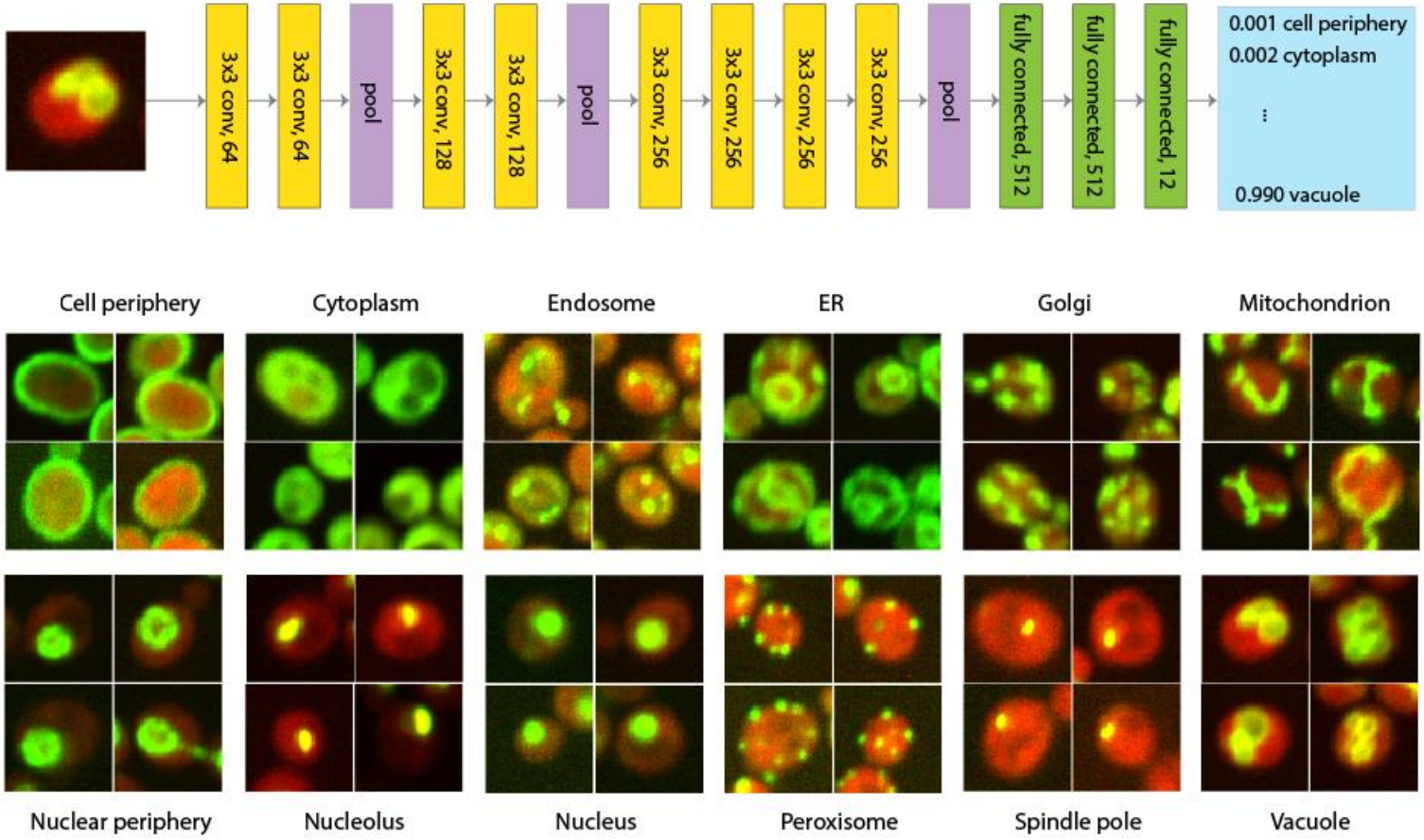


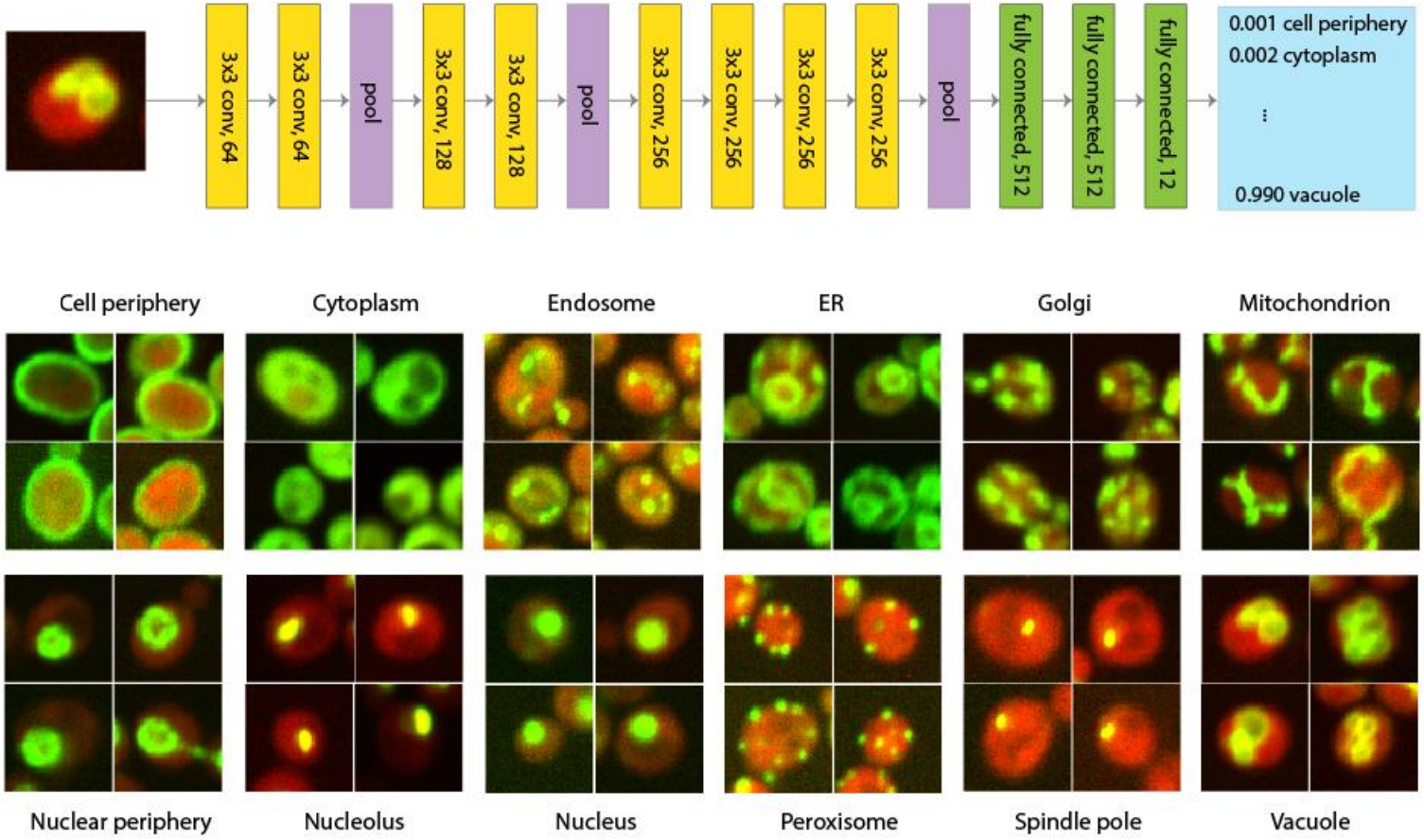
Deep learning for computational biology

Leopold Parts

18/09/2022

wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut



Deep learning for computational biology

Intuition, overview+ideas, tips
Training from Dmytro Fishman

Leopold Parts

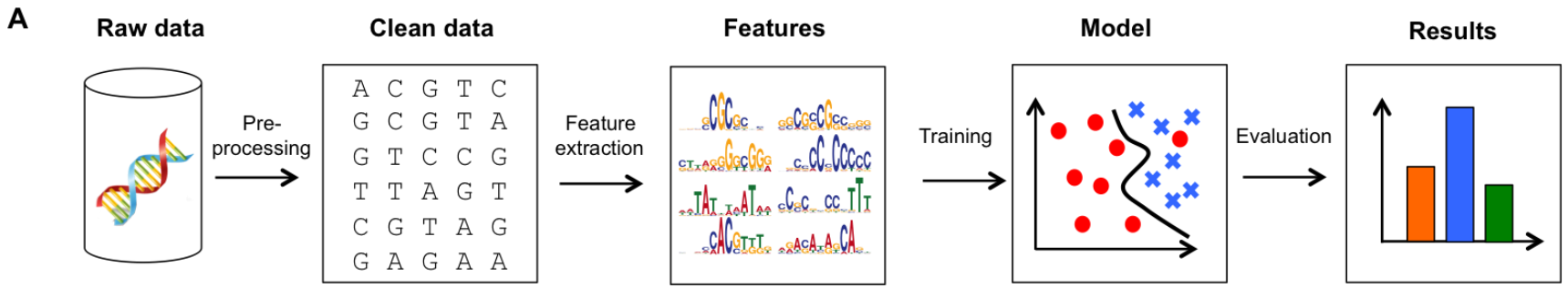
18/09/2022

wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

Why deep learning?

Why deep learning? The canonical machine learning workflow

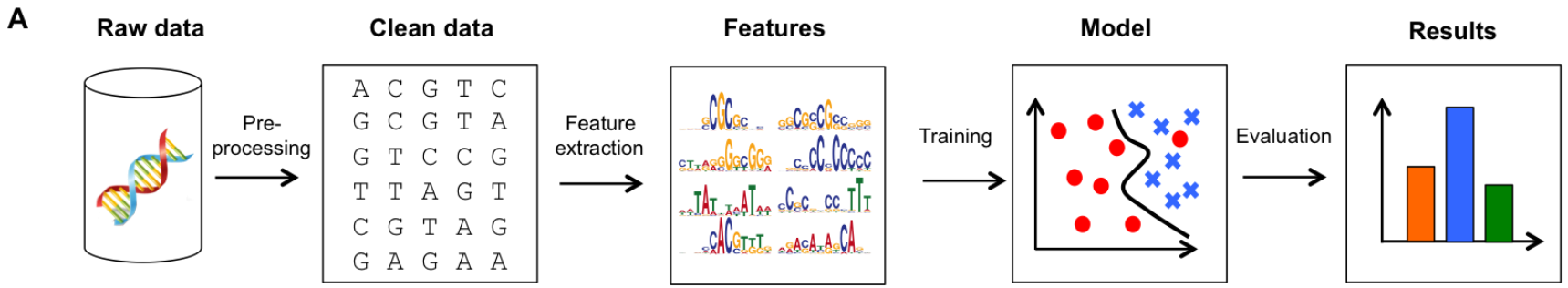


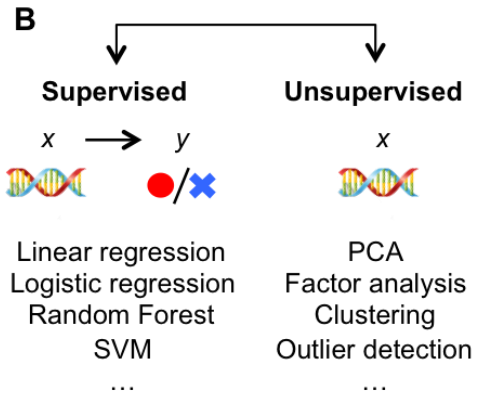
wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

From Angermuller*, Parnamaa*, Parts[$], Stegle[$] (Molecular Systems Biology)

Why deep learning? The canonical machine learning workflow



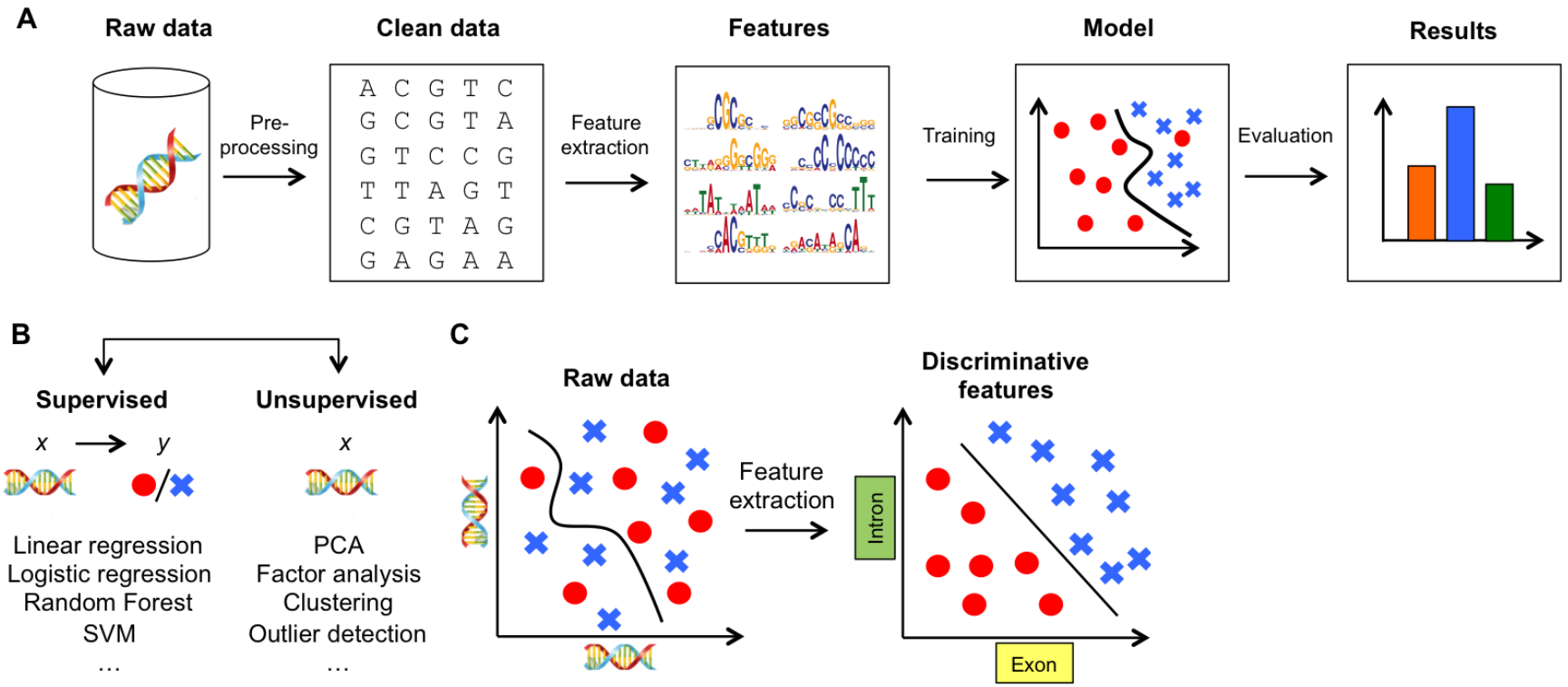


wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

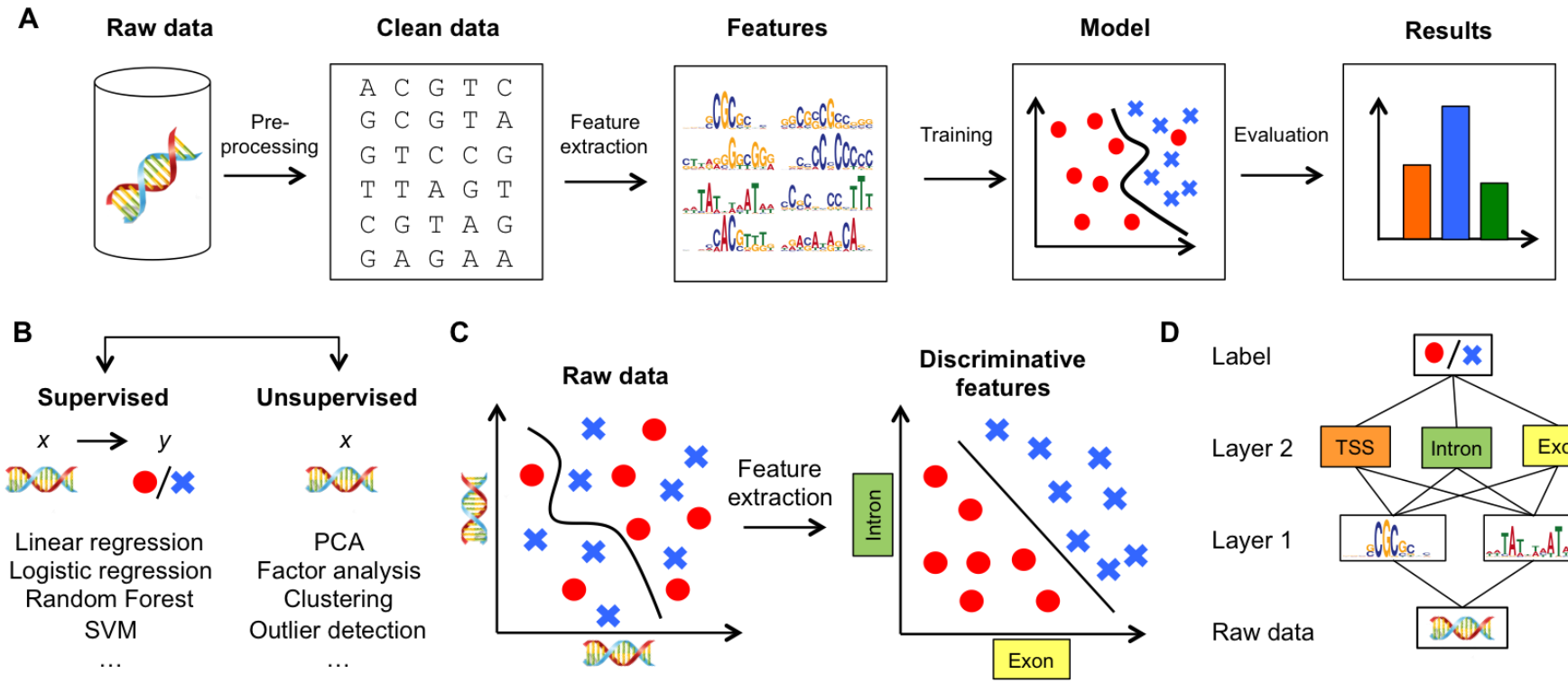
From Angermuller*, Parnamaa*, Parts[$], Stegle[$] (Molecular Systems Biology)

Why deep learning? The canonical machine learning workflow



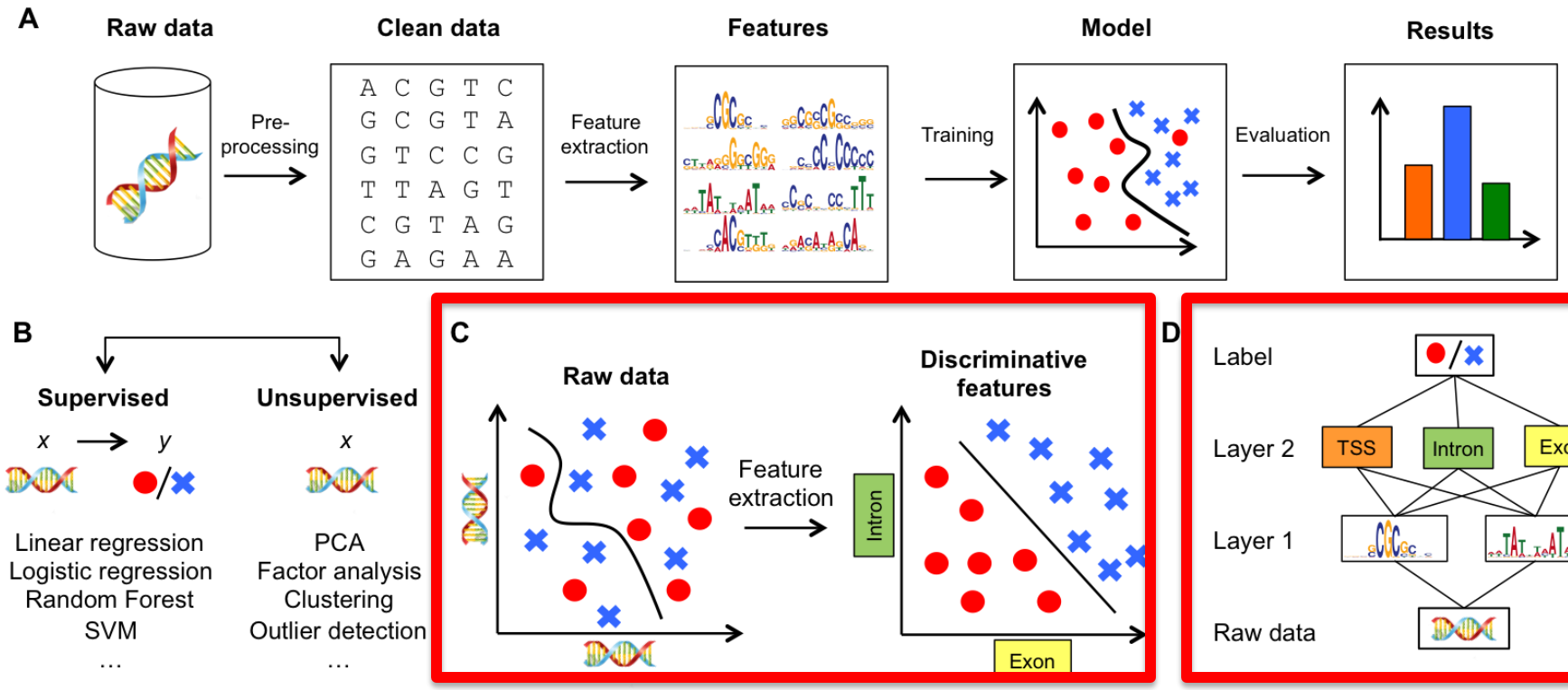
From Angermuller*, Parnamaa*, Parts$, Stegle$
(Molecular Systems Biology)

Why deep learning? The canonical machine learning workflow



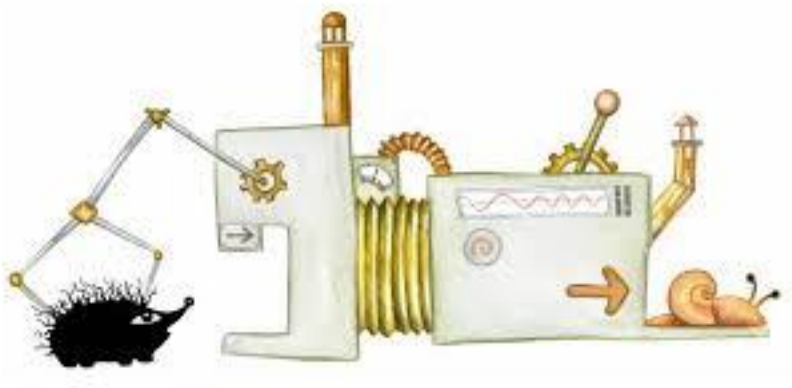
From Angermuller*, Parnamaa*, Parts$, Stegle$ (Molecular Systems Biology)

Why deep learning? The canonical machine learning workflow



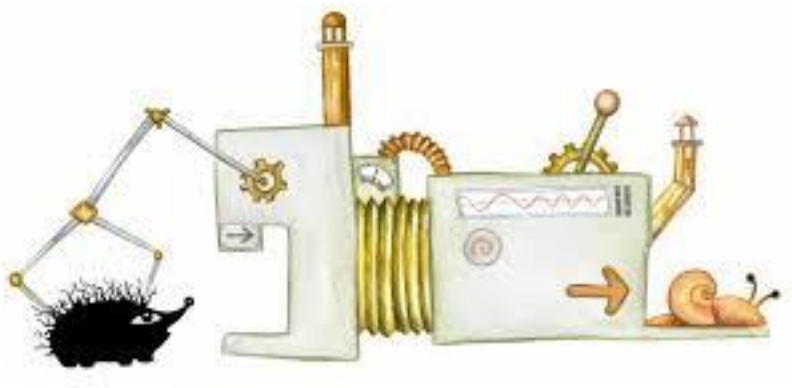
From Angermuller*, Parnamaa*, Parts$, Stegle$ (Molecular Systems Biology)

Deep neural networks - basics



“A mathematical function”

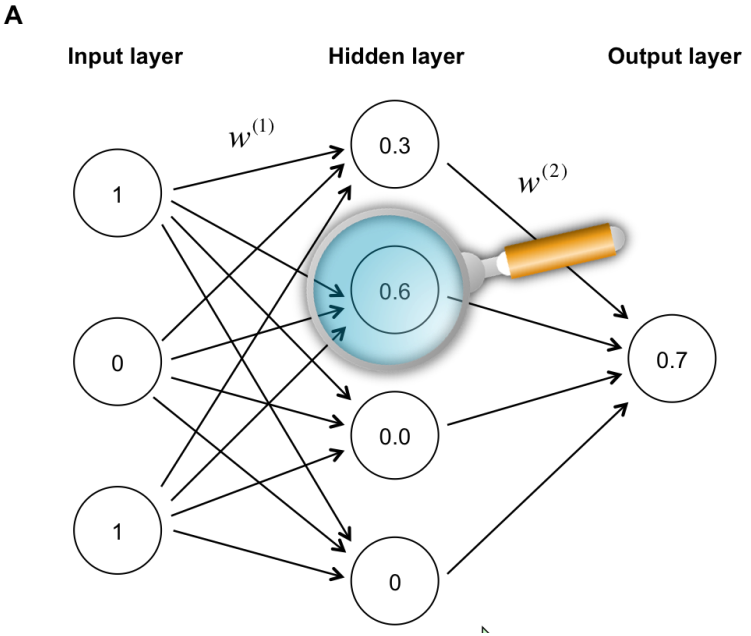
Deep neural networks - basics



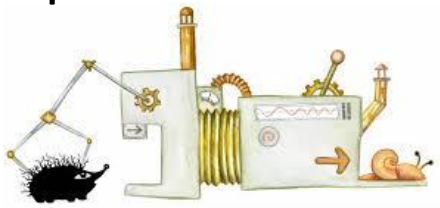
"A mathematical function"

Deep neural networks - basics

Neural network is a composite (hierarchical) mathematical function



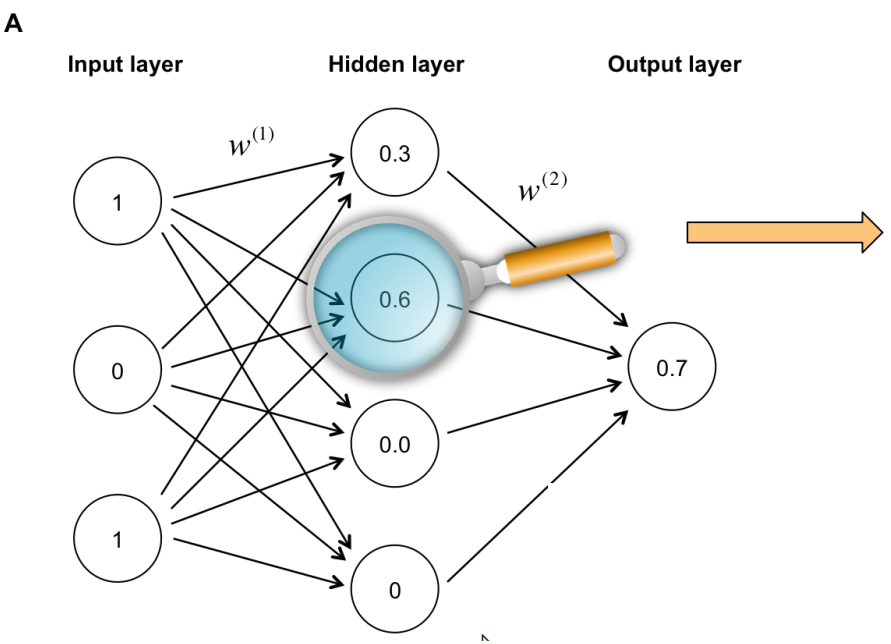
Example network structure

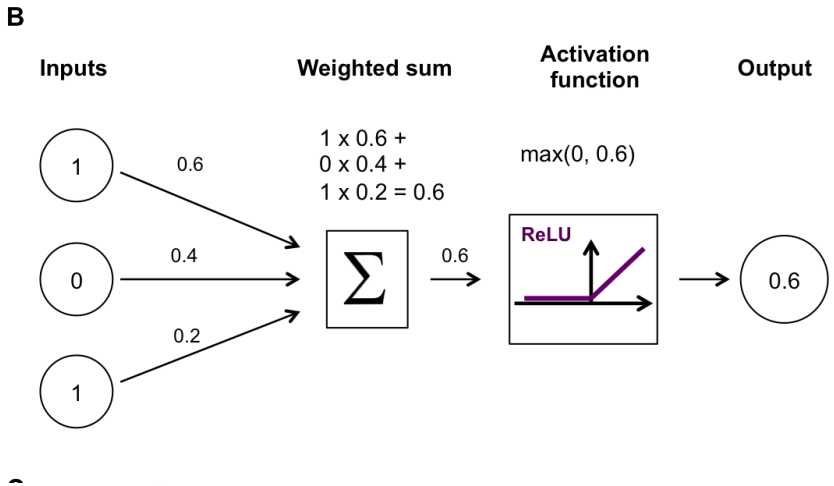


From Angermuller*, Parnamaa*, Parts$, Stegle$
(Molecular Systems Biology)

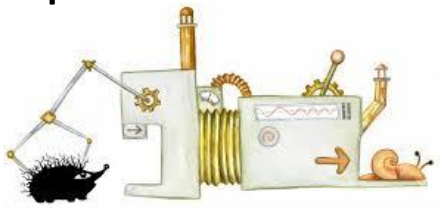
Deep neural networks - basics

Neural network is a composite (hierarchical) mathematical function

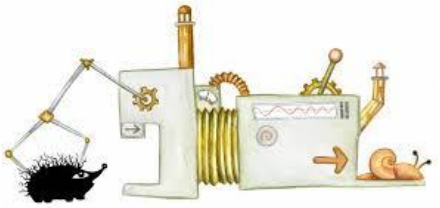




Example network structure



Example neuron workings



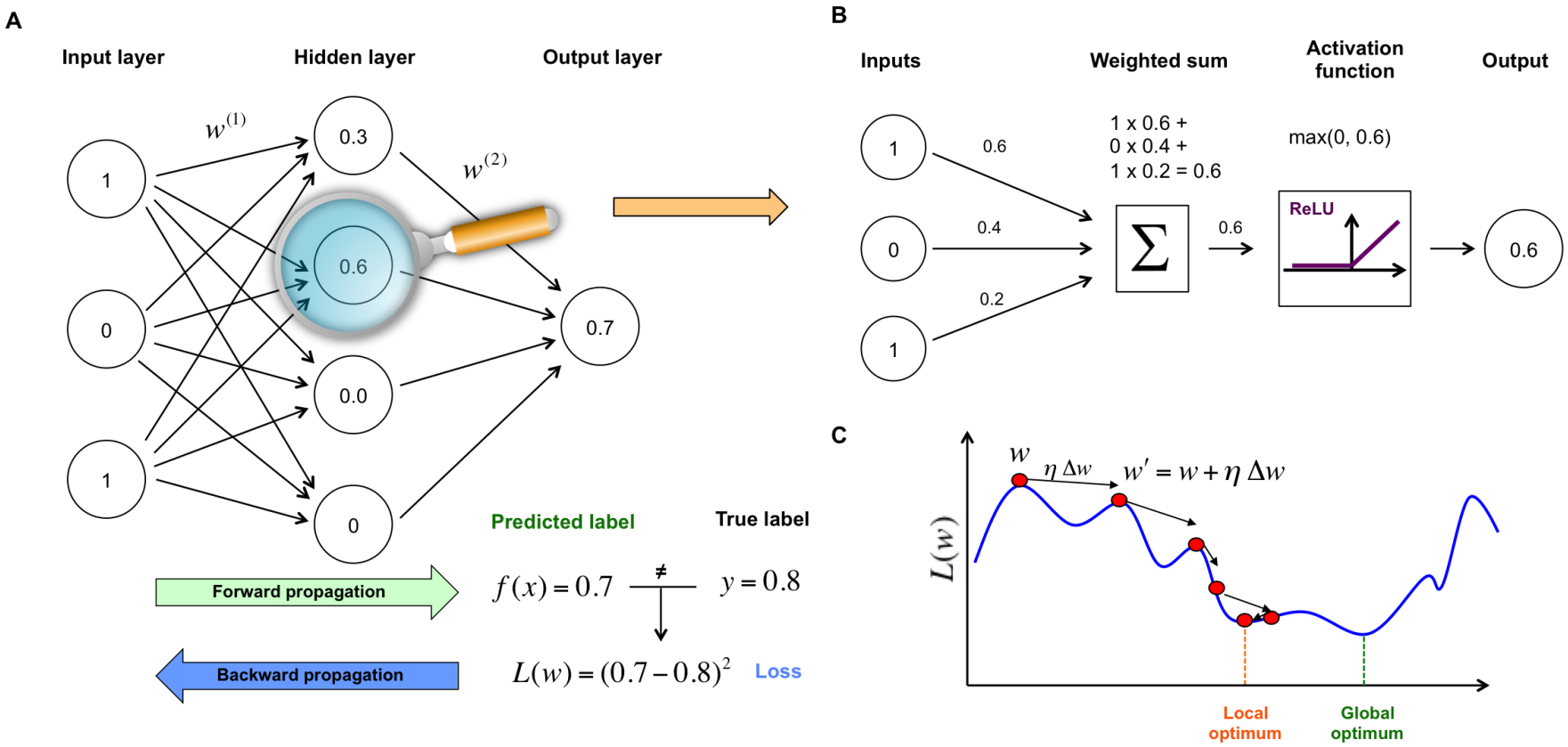
wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

From Angermuller*, Parnamaa*, Parts$, Stegle$
(Molecular Systems Biology)

Deep neural networks - basics

Neural network is a composite (hierarchical) mathematical function



Learning in the network = optimizing output loss by changing weights

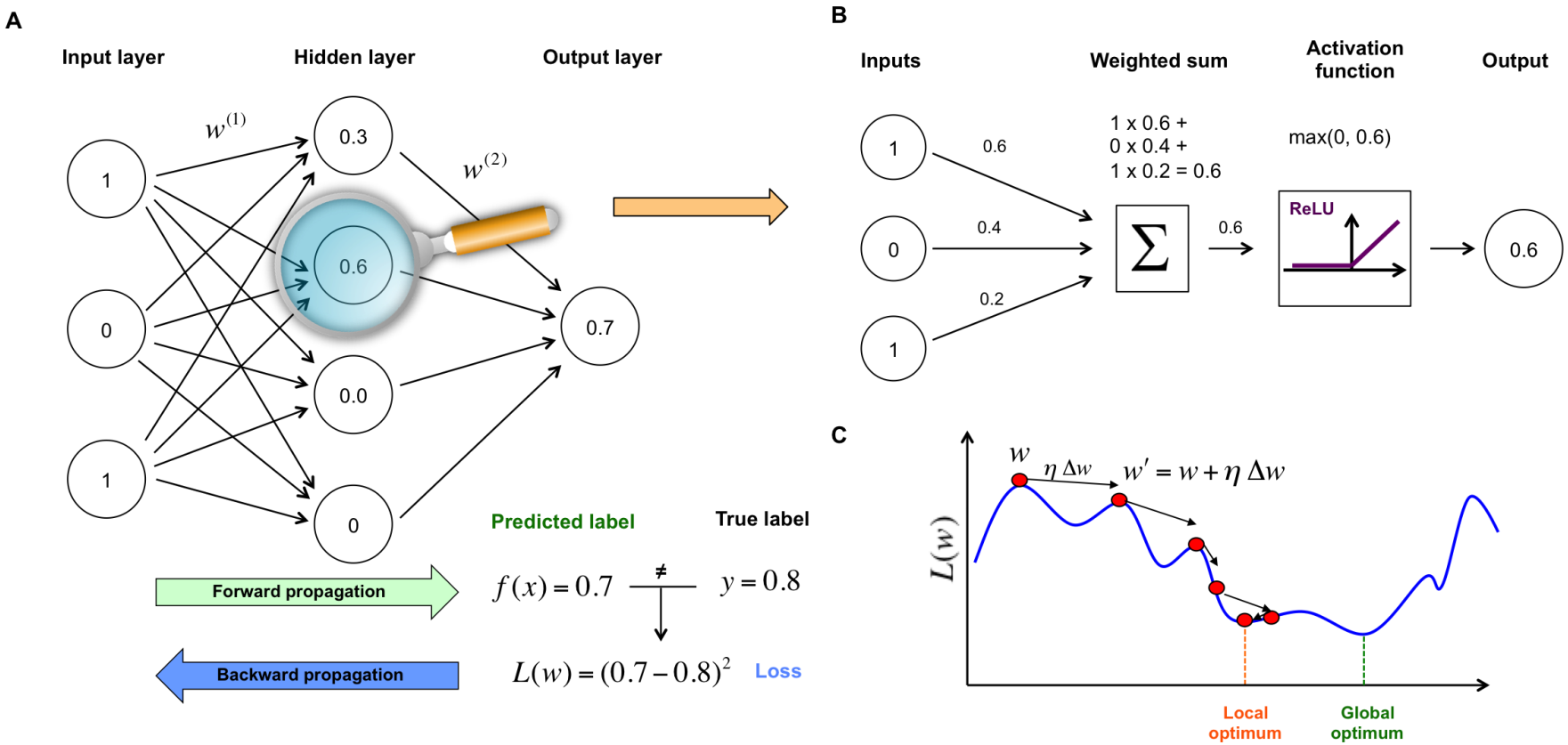
wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

From Angermuller*, Parnamaa*, Parts$, Stegle$
(Molecular Systems Biology)

Deep neural networks - basics

Neural network is a composite (hierarchical) ***mathematical function***



Learning in the network = **optimizing output loss** by changing weights

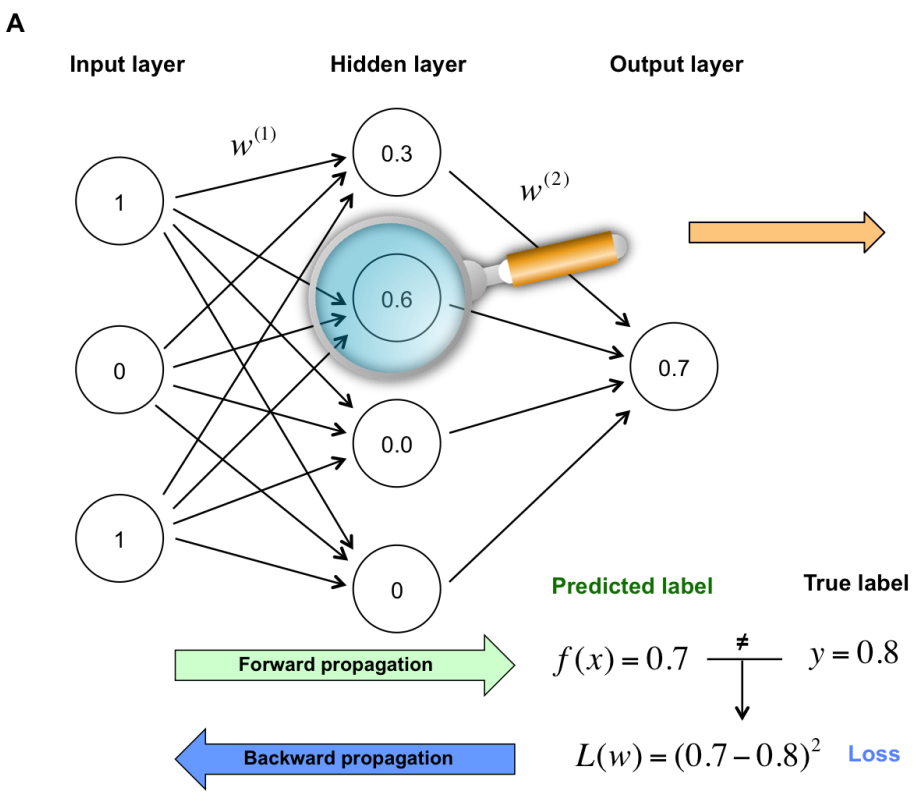
wellcome trust sanger institute

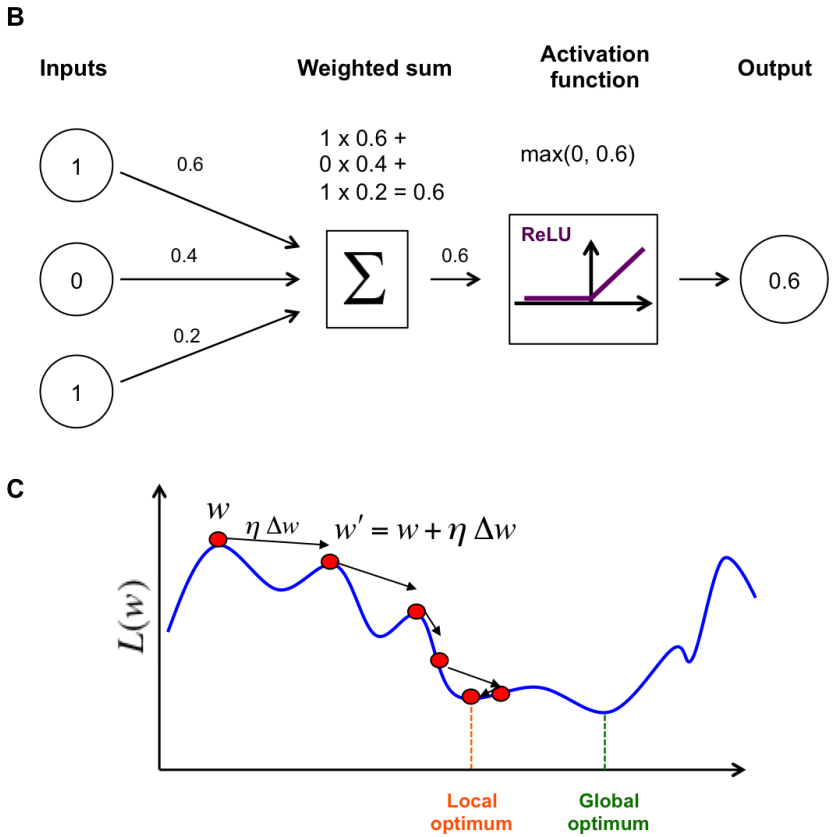
1632 TARTU ÜLIKOOL arvutiteaduse instituut

From Angermuller*, Parnamaa*, Parts$, Stegle$
(Molecular Systems Biology)

Deep neural networks - basics

Neural network is a composite (hierarchical) ***mathematical function***





Learning in the network = **optimizing output loss** by changing weights

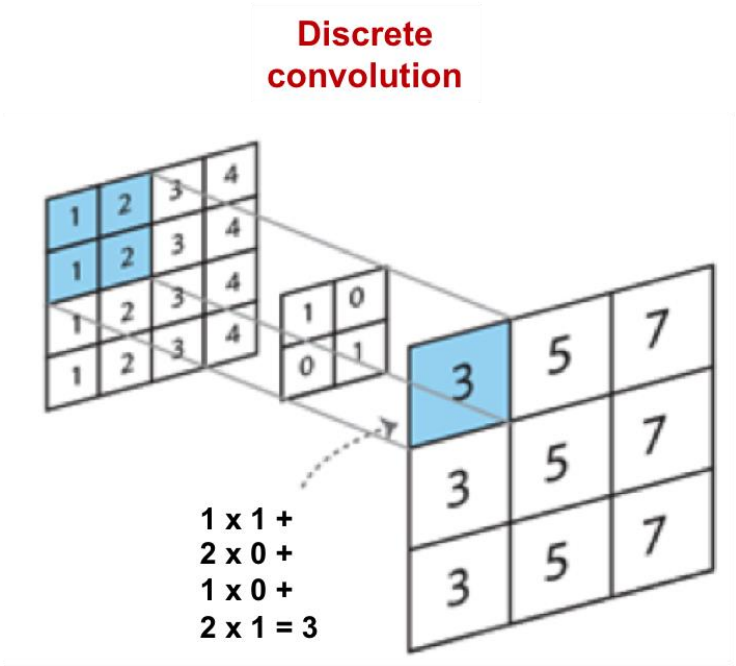
Building a network via composition and abstraction = **programming**

wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

From Angermuller*, Parnamaa*, Parts$^{\$}$, Stegle$^{\$}$ (Molecular Systems Biology)

Deep neural networks – pooling and convolution

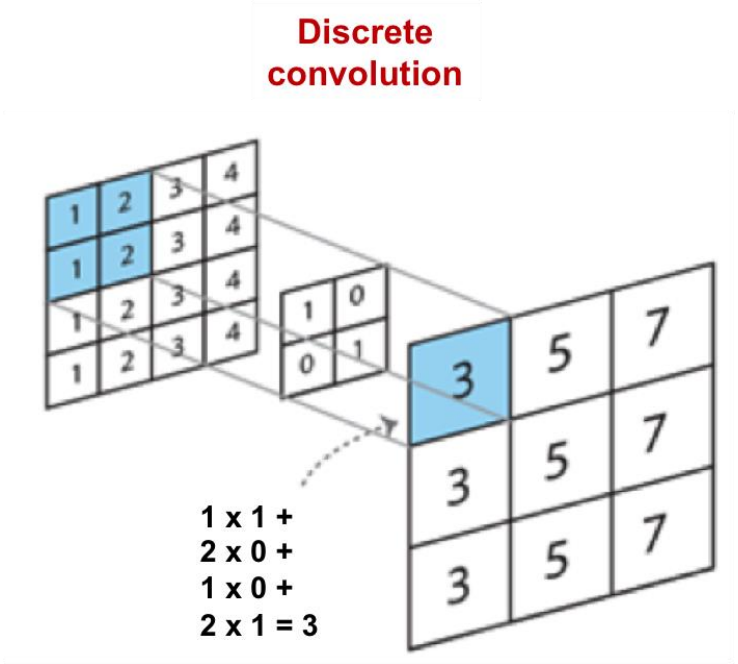


Strength of pattern match

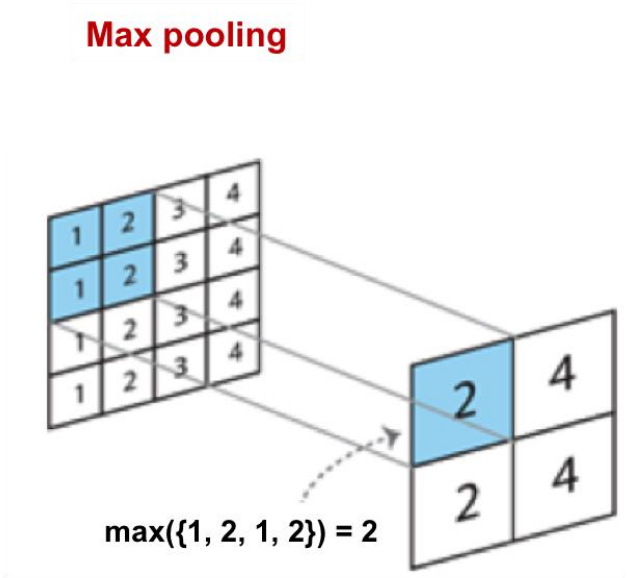
wellcome trust sanger institute
TARTU ÜLIKOOL arvutiteaduse instituut 1632

From Angermuller*, Parnamaa*, Parts$, Stegle$
(Molecular Systems Biology)

Deep neural networks – pooling and convolution



Strength of pattern match



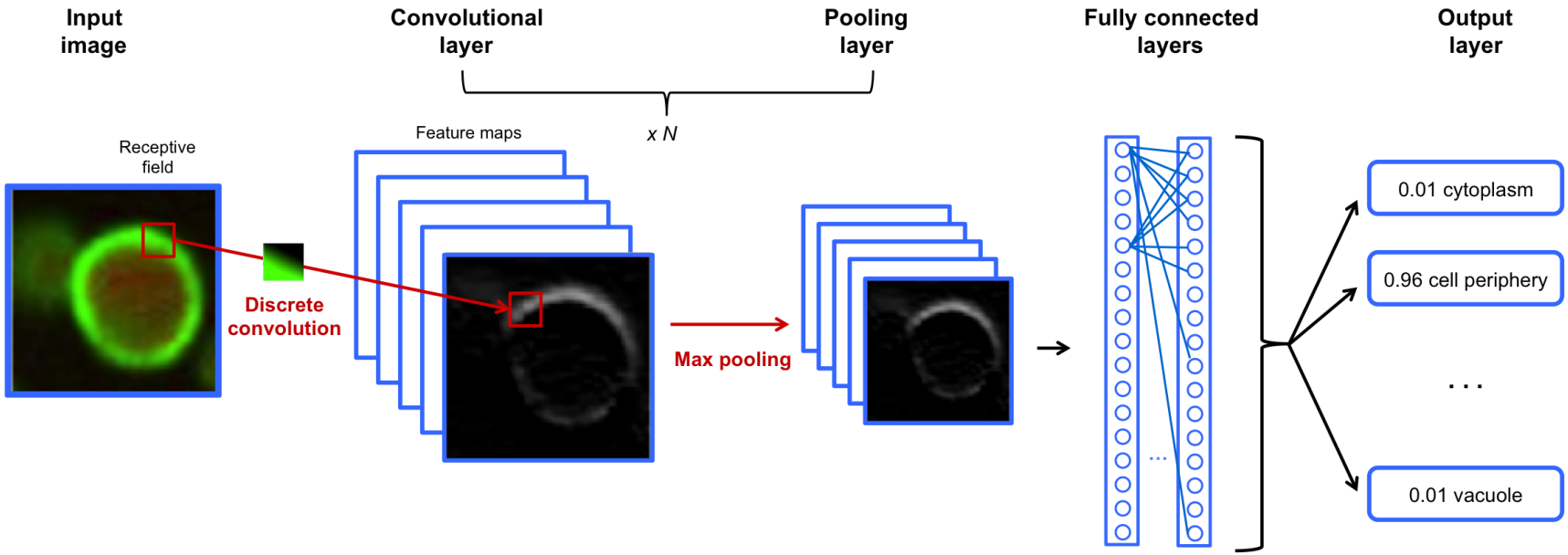
Pattern match in a large region

wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

From Angermuller*, Parnamaa*, Parts$, Stegle$
(Molecular Systems Biology)

Deep neural networks – pooling and convolution



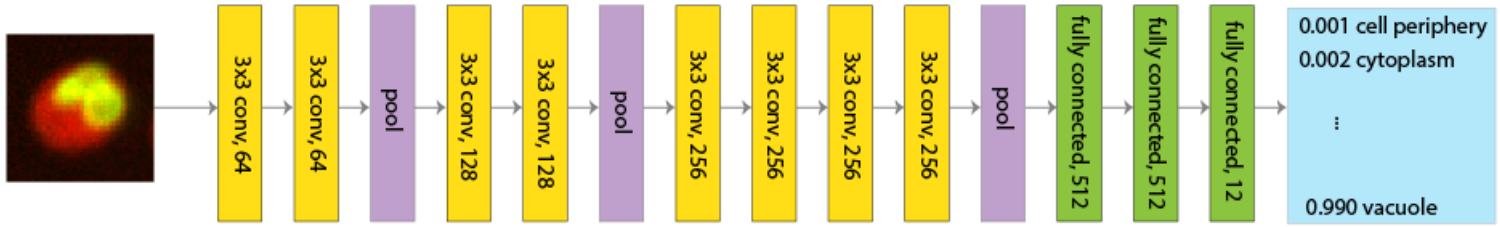
Composition of learning (many) features, and quantifying their presence in increasingly large contexts with a task-specific aggregation as output

From Angermuller*, Parnamaa*, Parts$, Stegle$ (Molecular Systems Biology)

End of Part 1. Deep neural networks - basics

- Neural network is a composite mathematical function of fixed form that implicitly captures a representation of the data
- Network model is fit by optimizing the flexible weights against a loss that explicitly measures what we value about network performance
- Building a network and choosing a loss are not unreasonably thought of as general purpose programming for capturing data and task representations

Aims: intuition, overview+ideas, tips



Part 2. Deep learning for …

Input: instance **x**
Target: property **y**
Representation?
Loss function?
Covariance function?

Part 2. Deep learning for …

Input: instance **x**

Target: property **y**

Representation?

- What features matter? How do we think about the problem?

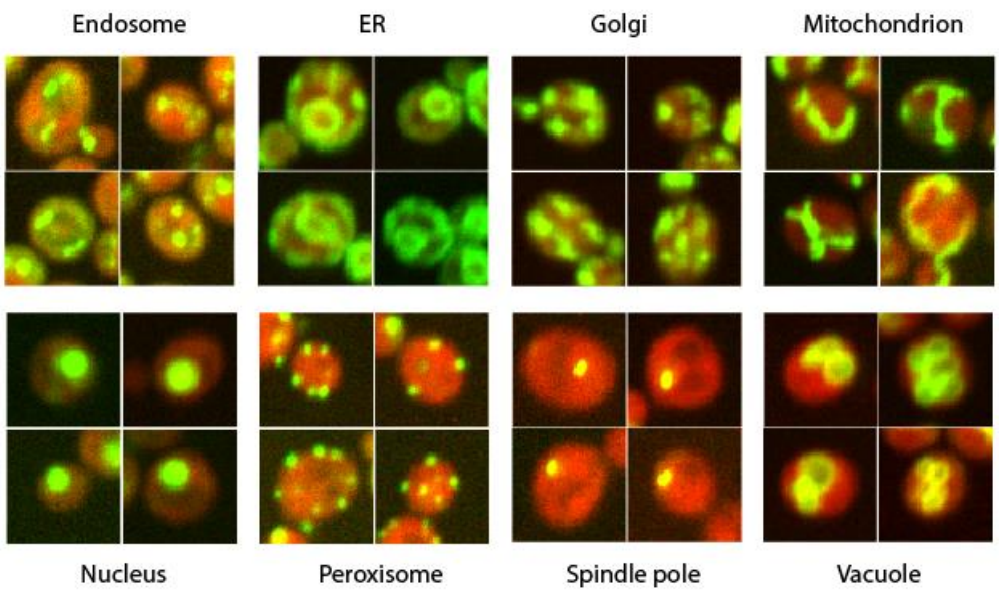
Loss function?

- What performance matters? Get what you train for…

Covariance function?

- What makes instances similar? Can use for training, clustering, kernel methods, …

Deep learning for subcellular localization



Input: 64x64 pixel image patch **x**

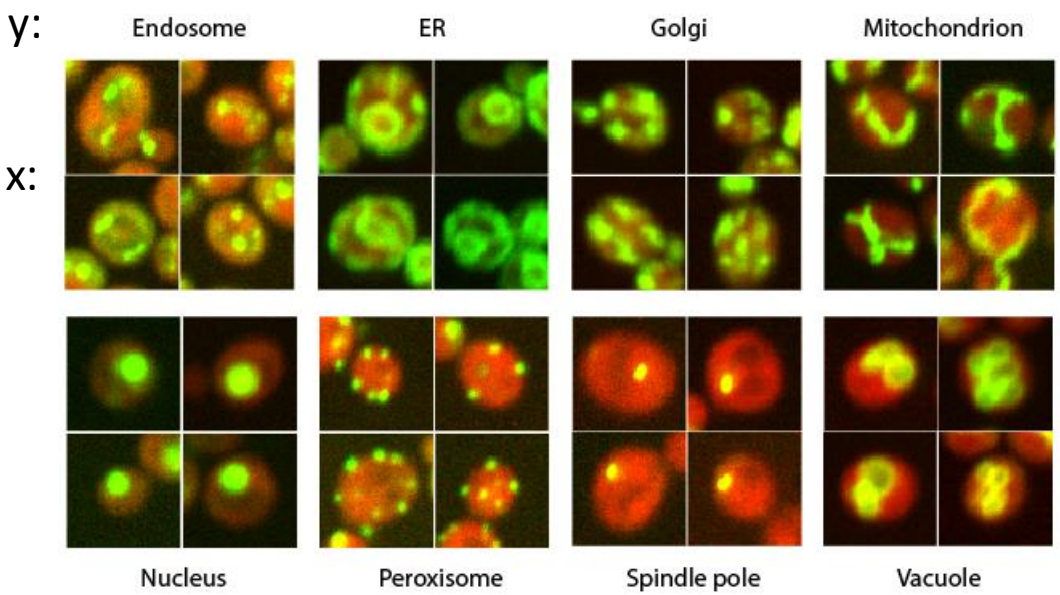
Target: class label **y**

Representation?

Loss function?

Covariance function?

Deep learning for subcellular localization



Input: 64x64 pixel image patch **x**

Target: class label **y**

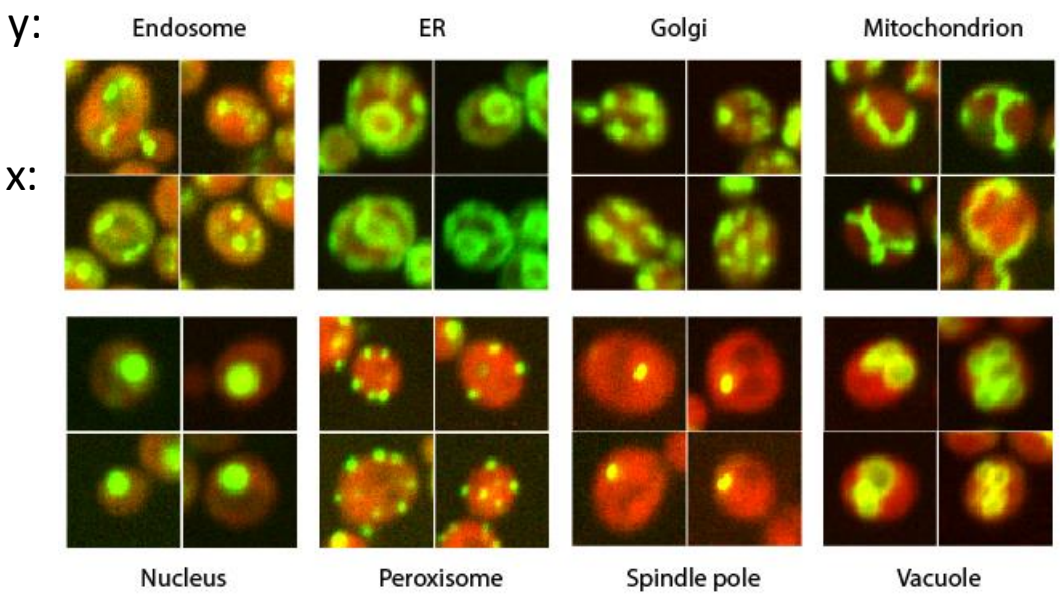
Representation

- Gabor filters, Zernike features, brightness, segmentation, …

Loss function?

Covariance function?

Deep learning for subcellular localization



Input: 64x64 pixel image patch **x**

Target: class label **y**

Representation

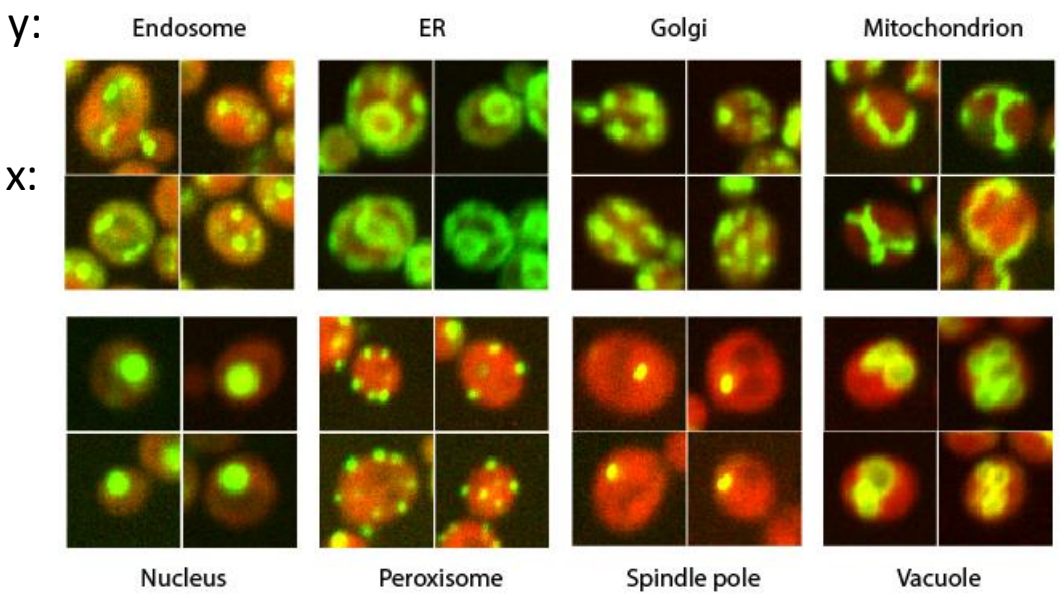
- Gabor filters, Zernike features, brightness, segmentation, …

Loss function

- Accuracy (balanced?)

Covariance function?

Deep learning for subcellular localization



Input: 64x64 pixel image patch **x**

Target: class label **y**

Representation

- Gabor filters, Zernike features, brightness, segmentation, …

Loss function

- Accuracy (balanced?)

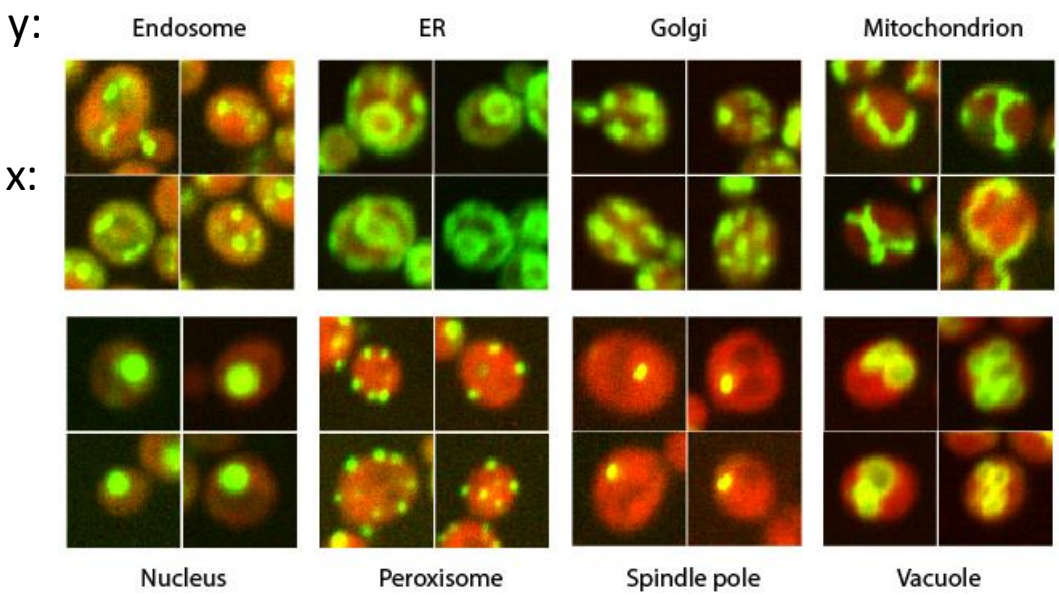
Covariance function

- E.g. from computed object properties

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for subcellular localization



Natural model: random forest (or these days, XGboost) of CellProfiler-extracted features

Input: 64x64 pixel image patch **x**

Target: class label **y**

Representation

- Gabor filters, Zernike features, brightness, segmentation, …

Loss function

- Accuracy (balanced?)

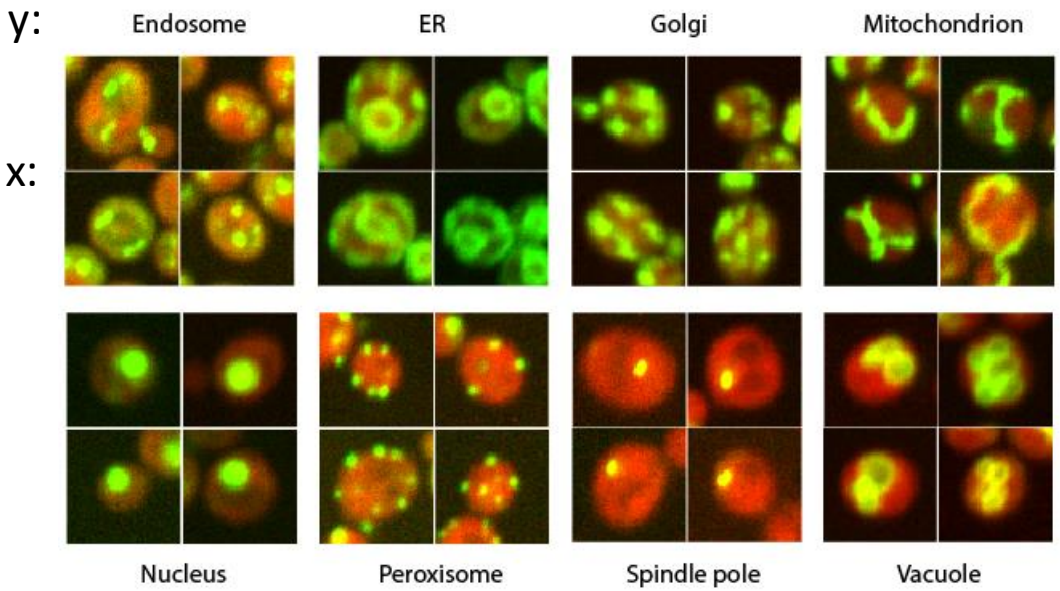
Covariance function

- E.g. from computed object properties

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for subcellular localization



Input: 64x64 patch **x**

Target: class label **y**

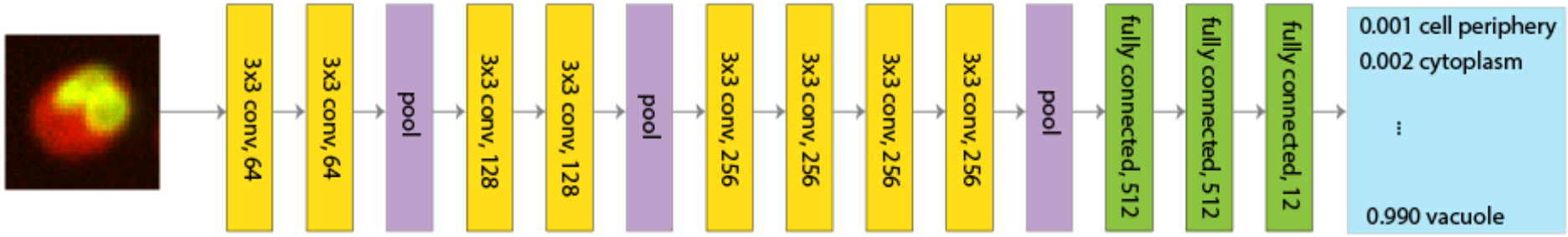
Representation

- Gabor filters, Zernike features, brightness, segmentation, …

Loss function

- Accuracy (balanced?)

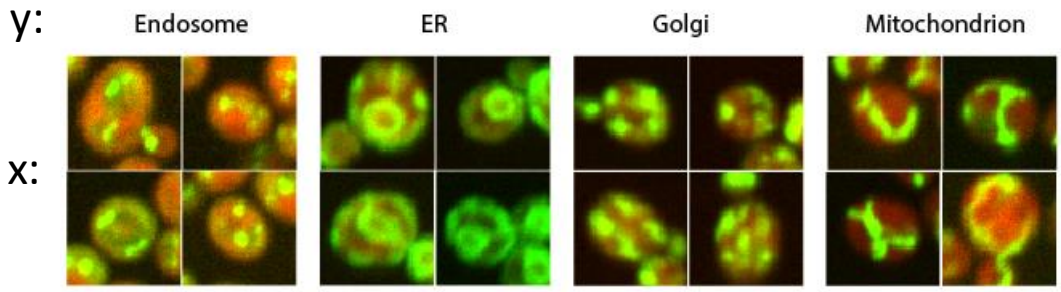
Covariance function



"DeepYeast" convolutional neural network classifier (>1,000,000 parameters)

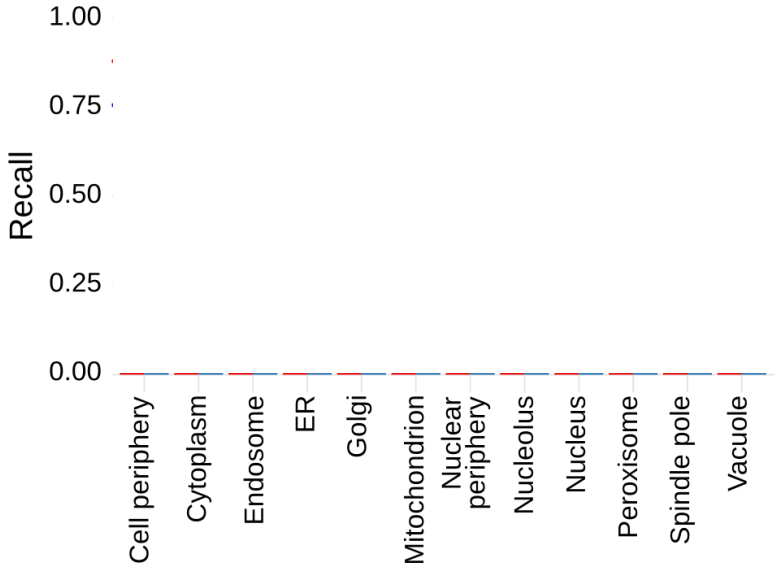
Tanel Pärnamaa

Deep learning for subcellular localization



Model performance

A



Input: 64x64 patch **x**

Target: class label **y**

Representation

- Gabor filters, Zernike features, brightness, segmentation, …

Loss function

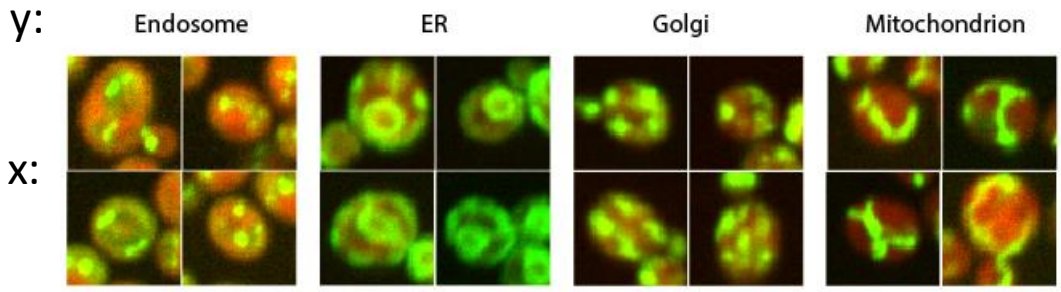
- Accuracy (balanced?)

Covariance function

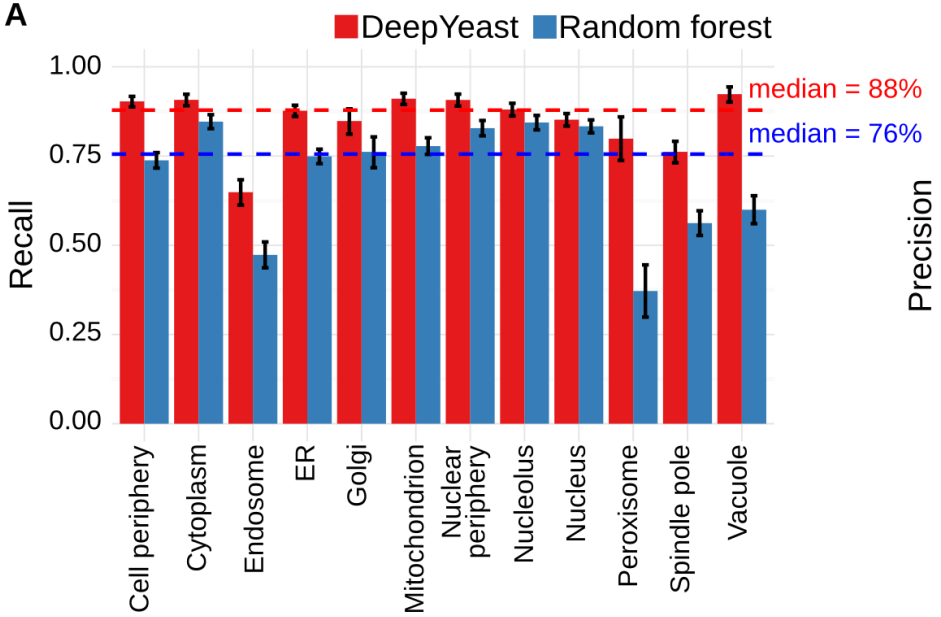
- E.g. from computed object properties

Tanel Pärnamaa

Deep learning for subcellular localization



Model performance



Input: 64x64 patch **x**

Target: class label **y**

Representation

- Gabor filters, Zernike features, brightness, segmentation, …

Loss function

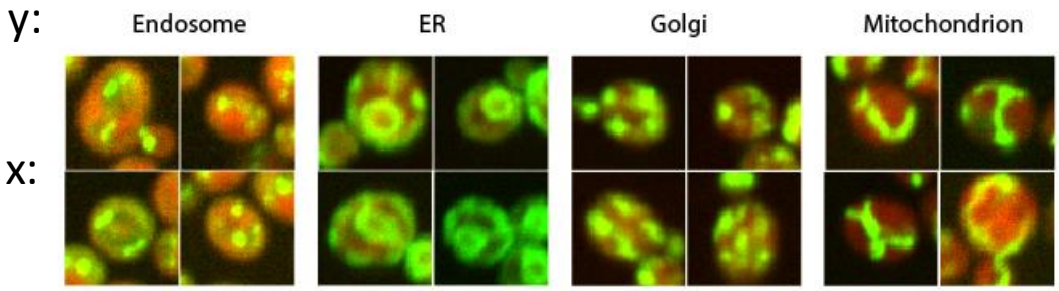
- Accuracy (balanced?)

Covariance function

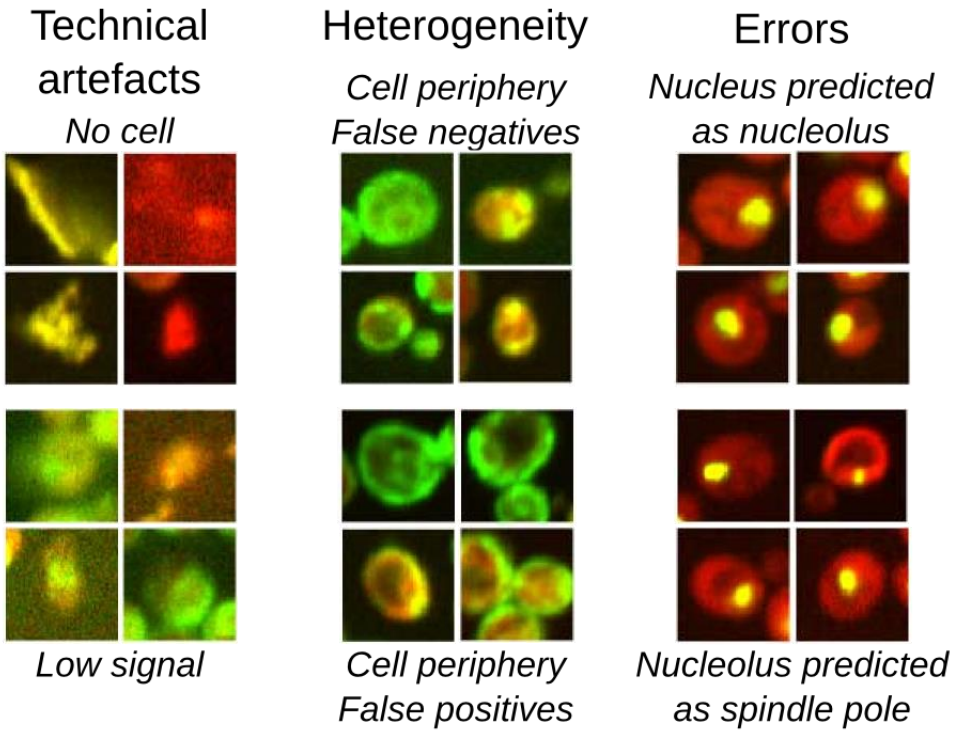
- E.g. from computed object properties

Tanel Pärnamaa

Deep learning for subcellular localization



Sources of errors



Input: 64x64 patch **x**

Target: class label **y**

Representation

- Gabor filters, Zernike features, brightness, segmentation, …

Loss function

- Accuracy (balanced?)

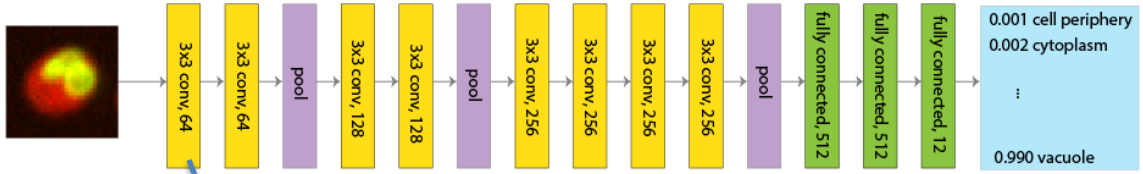
Covariance function

- E.g. from computed object properties

Tanel Pärnamaa

Deep learning for subcellular localization

Model interpretation



Five neurons from the layer

Three inputs that induce strongest neuron firing

Input: 64x64 patch **x**

Target: class label **y**

Representation

- Gabor filters, Zernike features, brightness, segmentation, …

Loss function

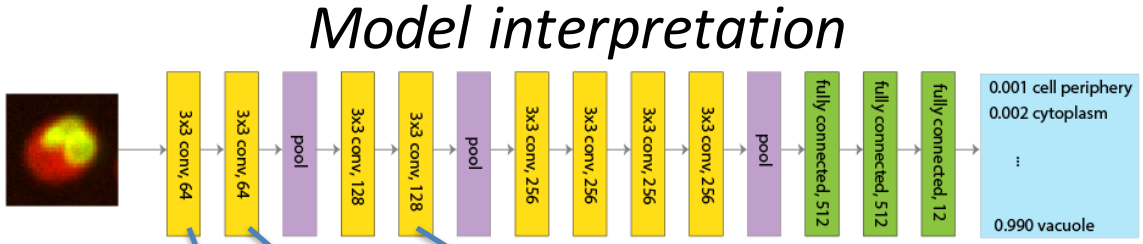
- Accuracy (balanced?)

Covariance function

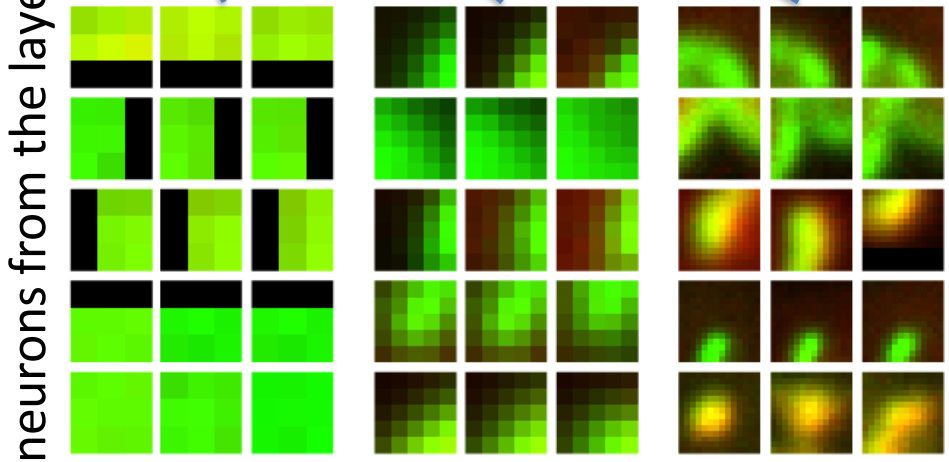
- E.g. from computed object properties

Tanel Pärnamaa

Deep learning for subcellular localization



Model interpretation



Five neurons from the layer

Three inputs that induce strongest neuron firing

Input: 64x64 patch **x**

Target: class label **y**

Representation

- Gabor filters, Zernike features, brightness, segmentation, …

Loss function

- Accuracy (balanced?)

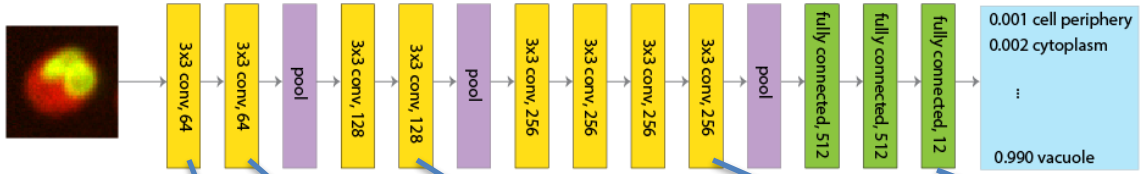
Covariance function

- E.g. from computed object properties

Tanel Pärnamaa

Deep learning for subcellular localization

Model interpretation



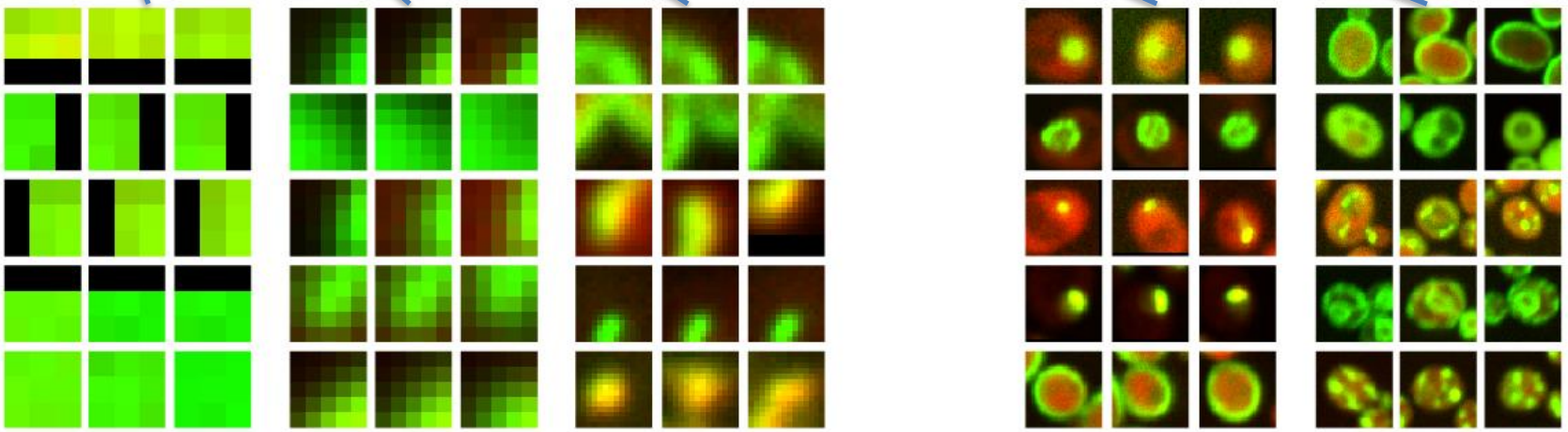
Input: 64x64 patch **x**

Target: class label **y**

Representation

- Gabor filters, Zernike features, brightness, segmentation, …

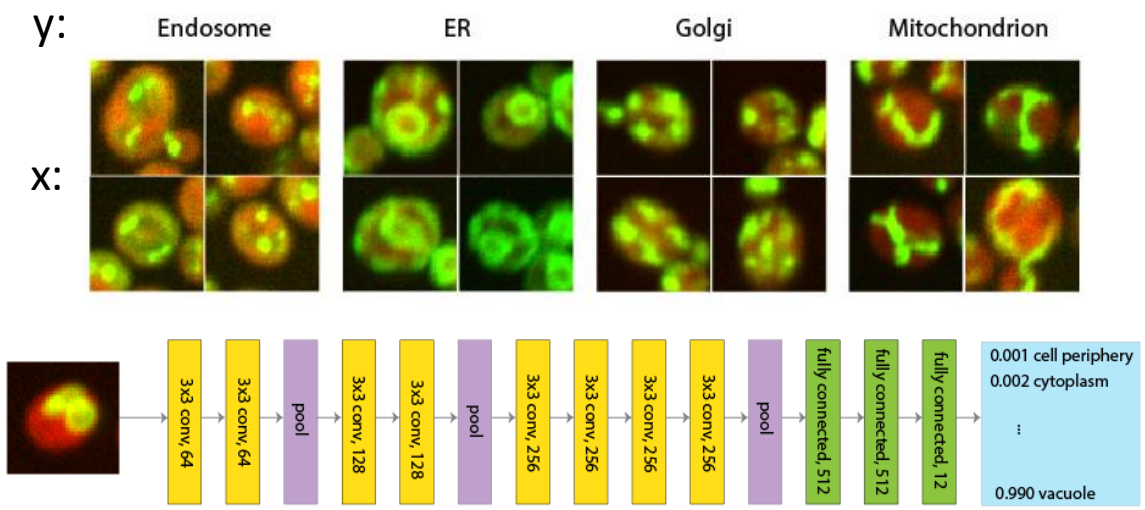
Five neurons from the layer



Three inputs that induce strongest neuron firing

Tanel Pärnamaa

Deep learning for subcellular localization



Input: 64x64 patch **x**

Target: class label **y**

Representation

- Gabor filters, Zernike features, brightness, segmentation, …

Loss function

- Accuracy (balanced?)

Covariance function

- E.g. from computed object properties

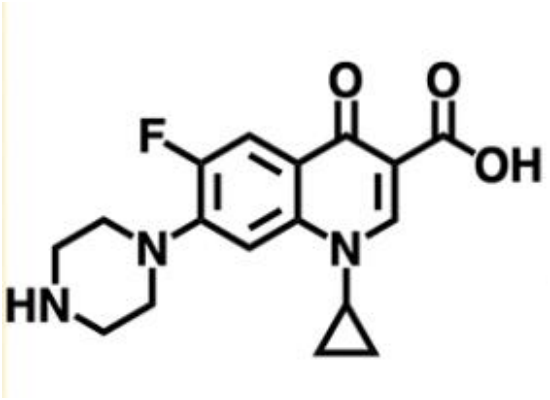
Key insights: convolutional networks are natural for local image features, pooling operations allow combining them across distance, and post-hoc activity analysis can help interpret. Dataset imbalance can be addressed in loss or training. Last errors often riddled with low quality data.

Example papers:
Accurate classification of protein subcellular localization from high throughput microscopy images using deep learning. Pärnamaa *et al.*, 2016.

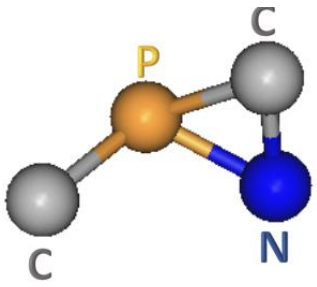
Tanel Pärnamaa

Deep learning in computational chemistry

x_1



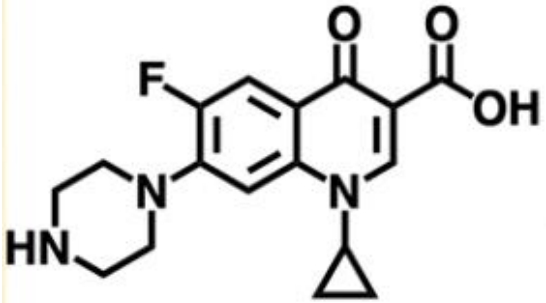
x_2



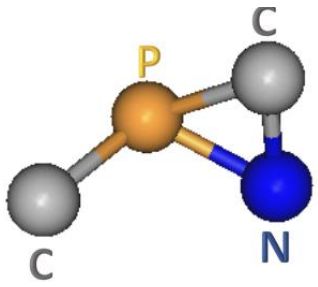
Input: molecule **x**
Target: property **y** (e.g. inhibits RAS protein, energy level)
Representation?
Loss function?
Covariance function?

Deep learning in computational chemistry

x_1



x_2



Input: molecule **x**

Target: property **y** (e.g. inhibits RAS protein, energy level)

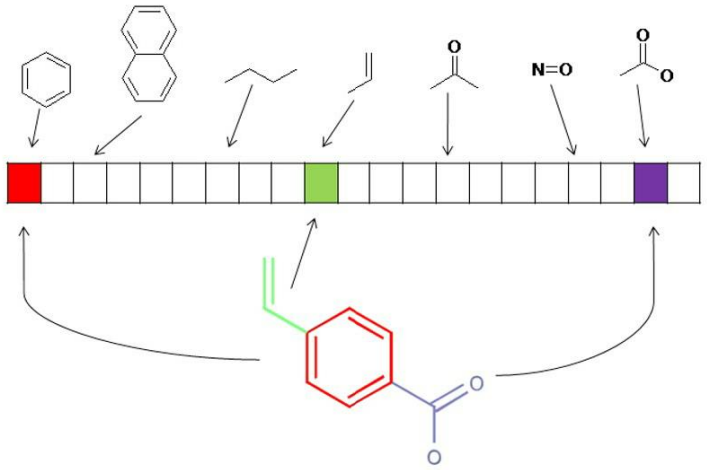
Representation

- Molecular fingerprinting

Loss function?

Covariance function?

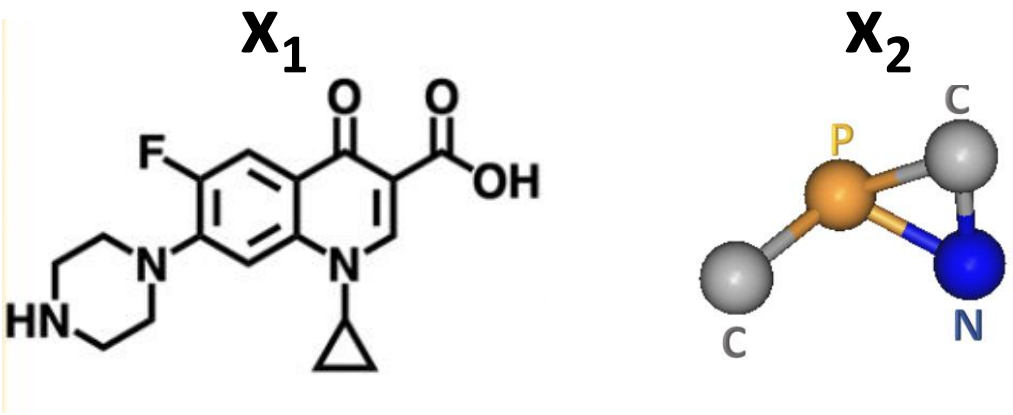
Molecular finger-printing



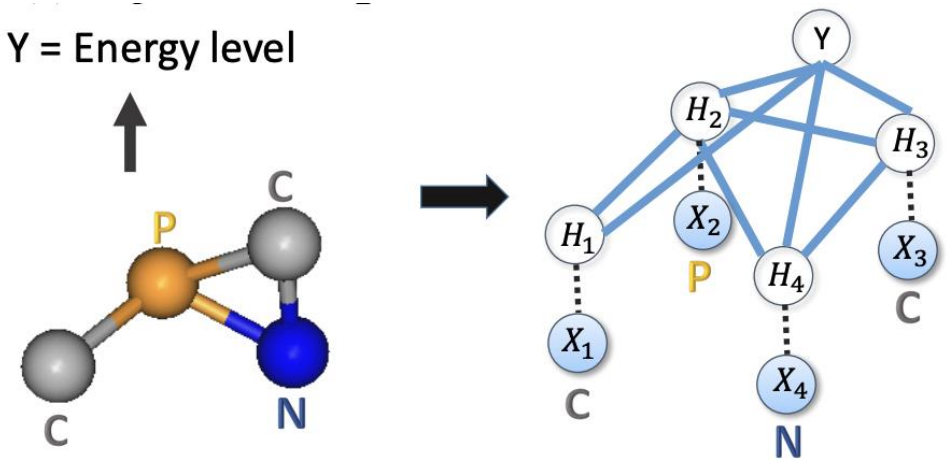
wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in computational chemistry



Learned representation



Input: molecule **x**

Target: property **y** (e.g. inhibits RAS, energy level)

Representation

- Molecular fingerprinting
- Learned vector

Loss function

- Autoencoding
- Property prediction

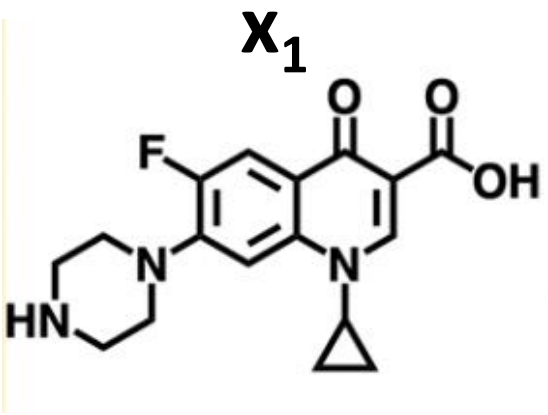
Covariance function?

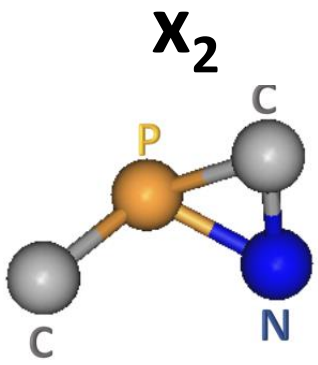
^

wellcome trust sanger institute

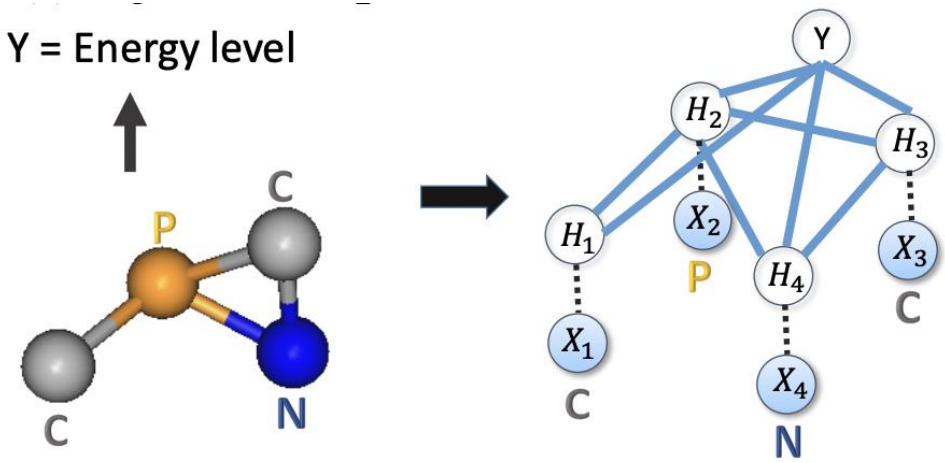
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in computational chemistry





Learned representation



Input: molecule **x**

Target: property **y** (e.g. inhibits RAS, energy level)

Representation

- Molecular fingerprinting
- Learned vector

Loss function

- Autoencoding
- Property prediction

Covariance function?

Two neural networks:

1) Embedding, to learn $h_i(\{x_j\})$ [e.g. molecular fingerprints]

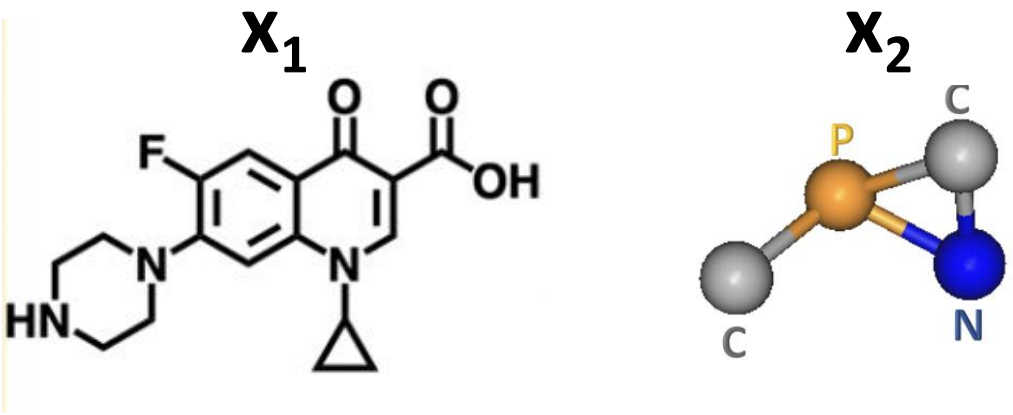
2) Pooling, to learn $\hat{y}(\{h_j\})$ [e.g. sigmoid($\mathbf{w}^t$ sum($\mathbf{h}_i$))]

Can learn jointly

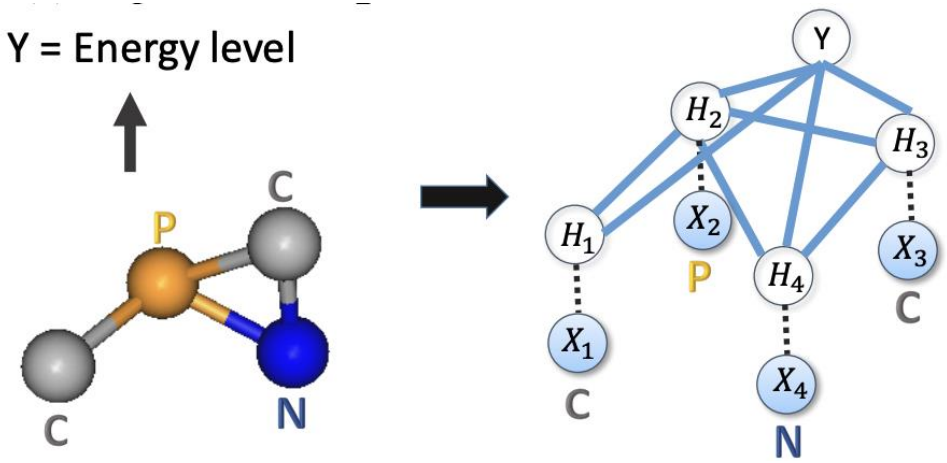
wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in computational chemistry



Learned representation



Input: molecule **x**

Target: property **y** (e.g. inhibits RAS, energy level)

Representation

- Molecular fingerprinting
- Learned vector

Loss function

- Autoencoding
- Property prediction

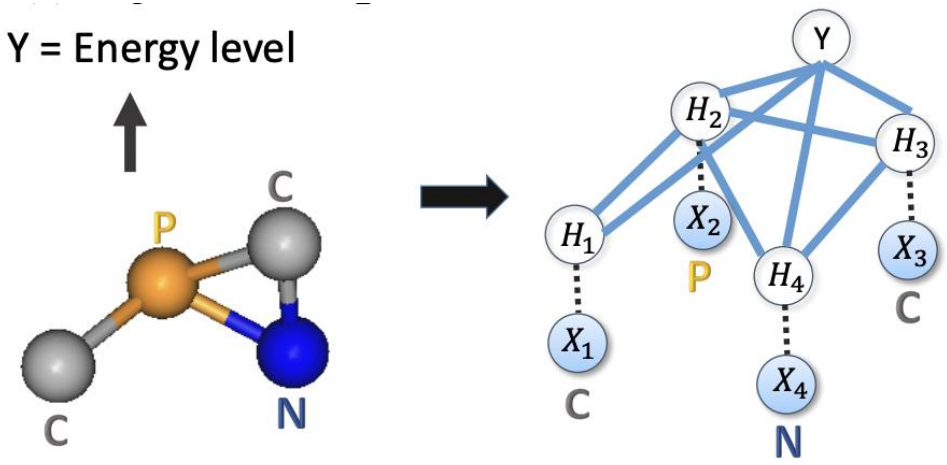
Covariance function

- Number of shared substructures
- Fisher kernel (based on model fit; similarity of gradient)

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in computational chemistry



Key insight: networks parametrize how 1) atom features and 2) properties of distant parts of the molecule influence 3) global properties. Each data point becomes a separate model that shares parameters with others.

Example papers: Discriminative Embeddings of Latent Variable Models for Structured Data. Dai *et al.*, MLR 2016
Analyzing Learned Molecular Representations for Property Prediction. Yang *et al.*, JCIM 2019.
Search Graph-NNs for molecule properties

Input: molecule **x**
Target: property **y** (e.g. inhibits RAS, energy level)
Representation

- Molecular fingerprinting
- Learned vector

Loss function

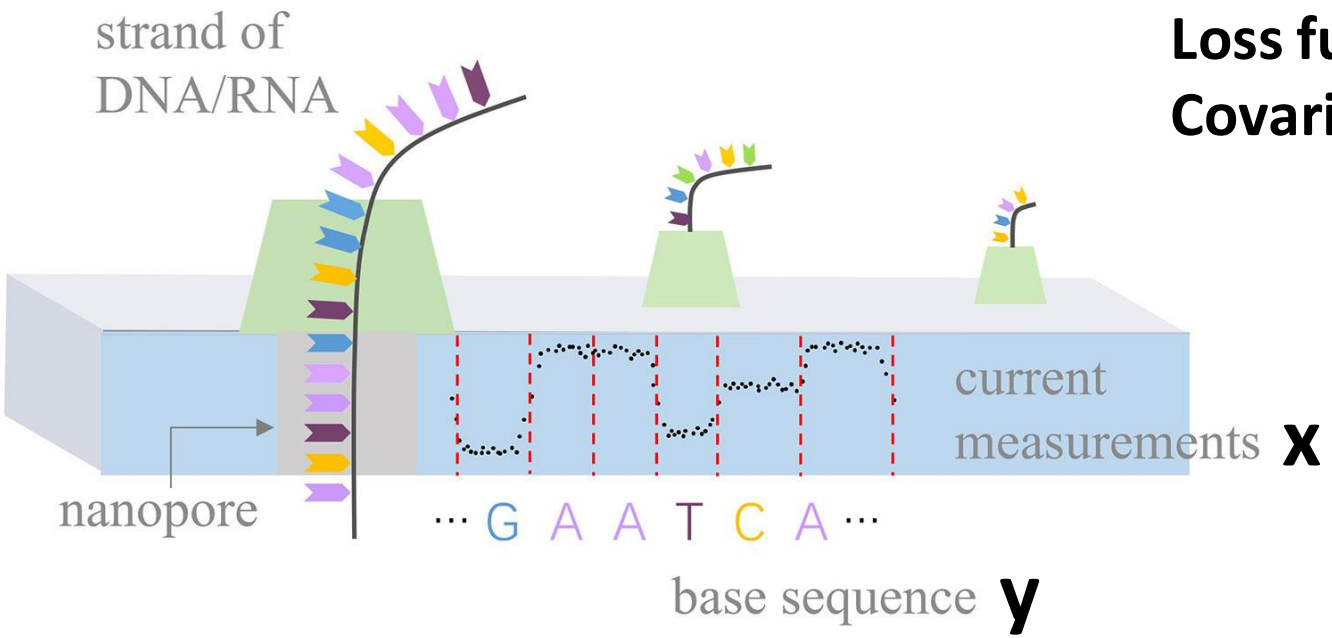
- Autoencoding
- Property prediction

Covariance function

- Number of shared substructures
- Fisher kernel (based on model fit; similarity of gradient)

wellcome trust sanger institute
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for DNA base calling (nanopore)



Input: current **x**

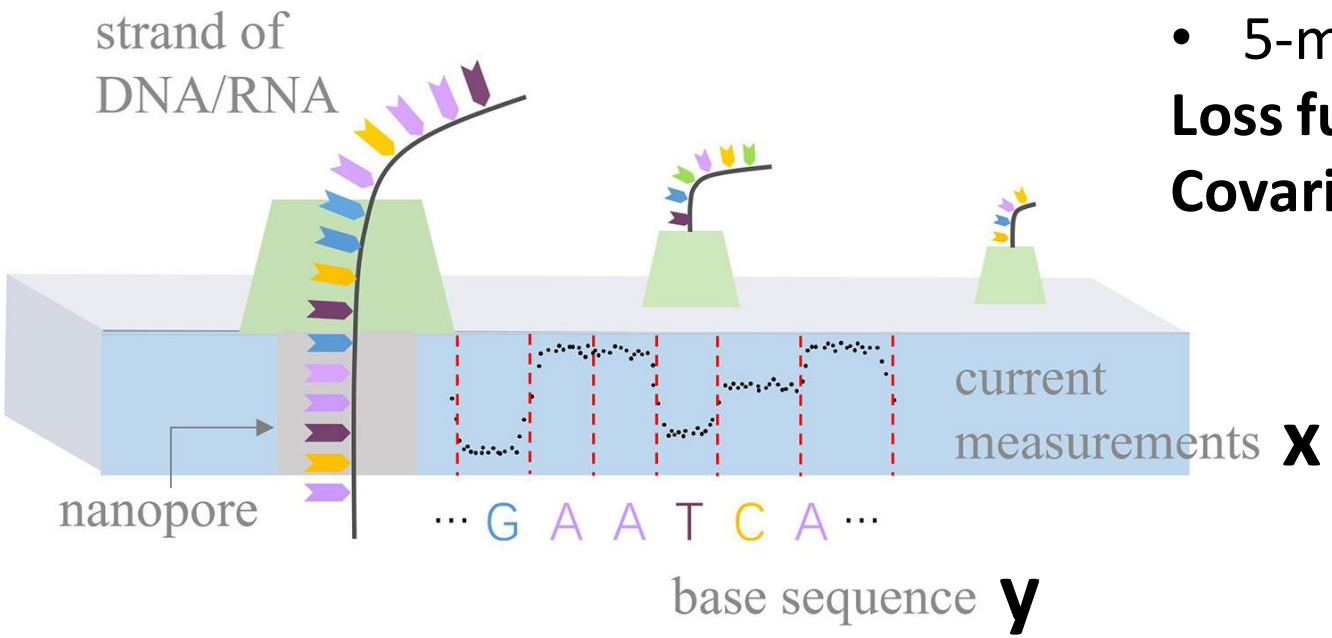
Target: sequence **y** (e.g. ACCATTGAAC)

Representation?

Loss function?

Covariance function?

Deep learning for DNA base calling (nanopore)



Input: current **x**

Target: sequence **y** (e.g. ACCATTGAAC)

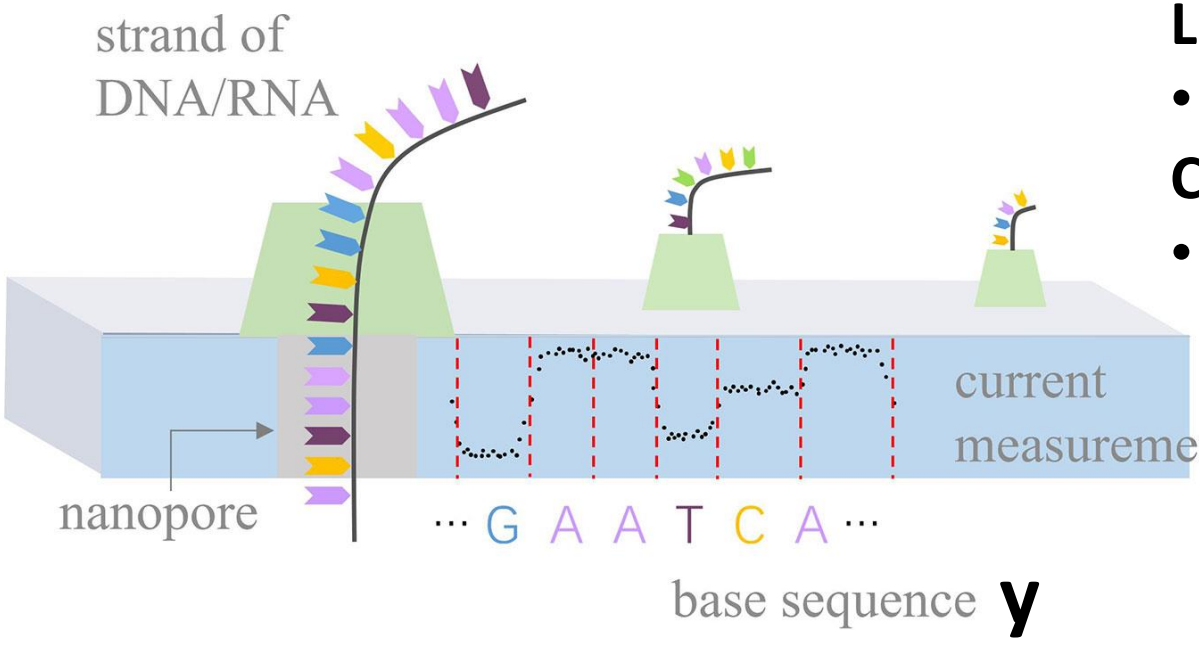
Representation

- 5-mer inside pore

Loss function?

Covariance function?

Deep learning for DNA base calling (nanopore)



Input: current **x**

Target: sequence **y** (e.g. ACCATTGAAC)

Representation

- 5-mer inside pore

Loss function

- Base calling error

Covariance function

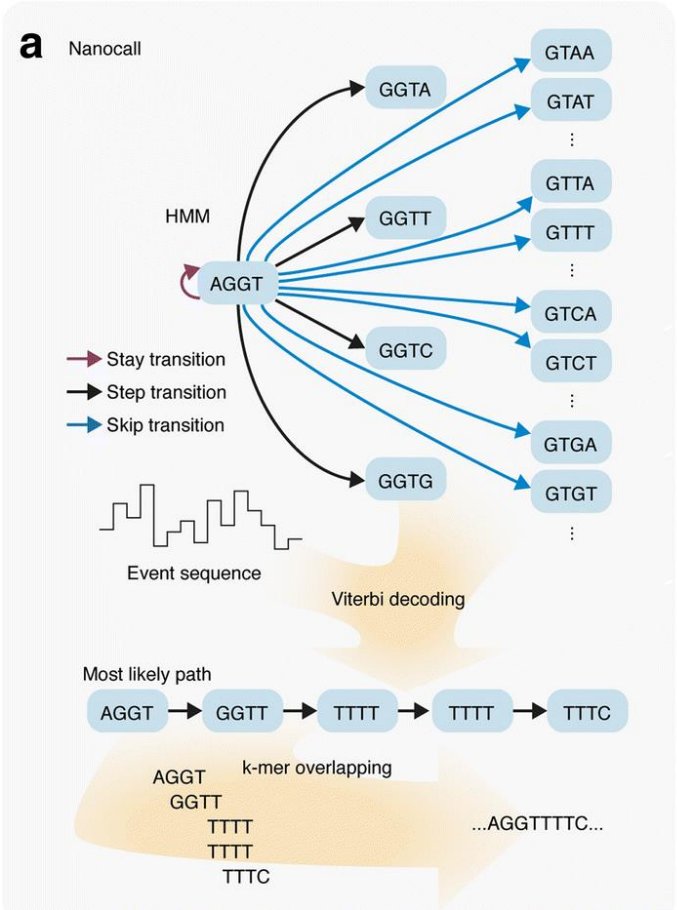
- Internal: linking currents across distance

wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

Deep learning for DNA base calling (nanopore)

A natural approach: hidden Markov model



Input: current **x**

Target: sequence **y** (e.g. ACCATTGAAC)

Representation

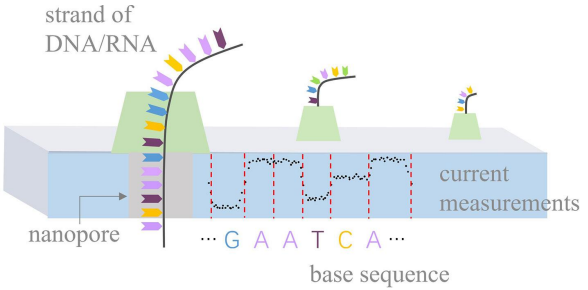
- 5-mer inside pore

Loss function

- Base calling error

Covariance function

- Internal: linking currents across distance

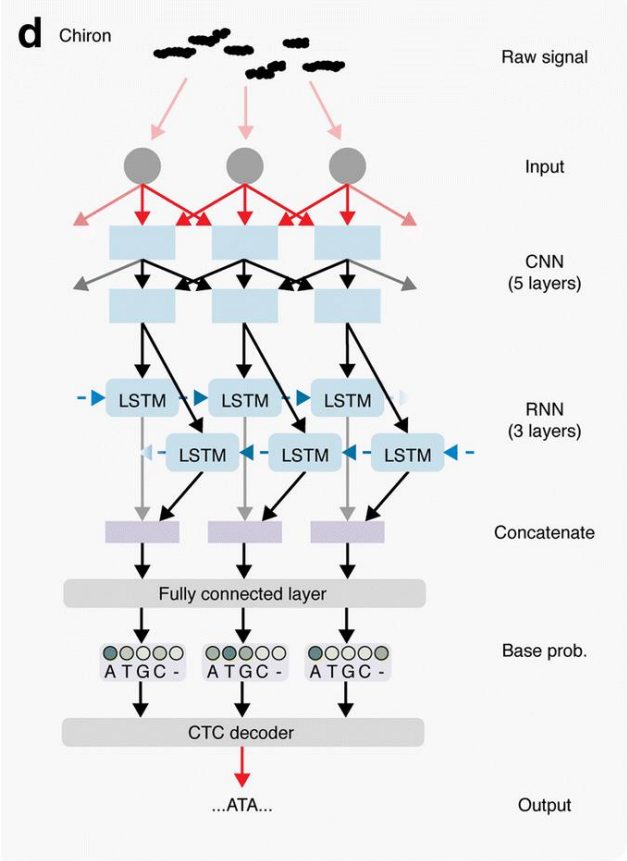


wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for DNA base calling (nanopore)

A relaxed approach: convolutional model



Input: current **x**

Target: sequence **y** (e.g. ACCATTGAAC)

Representation

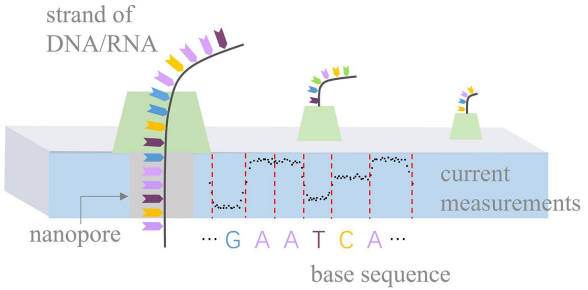
- 5-mer inside pore

Loss function

- Base calling error

Covariance function

- Internal: linking currents across distance

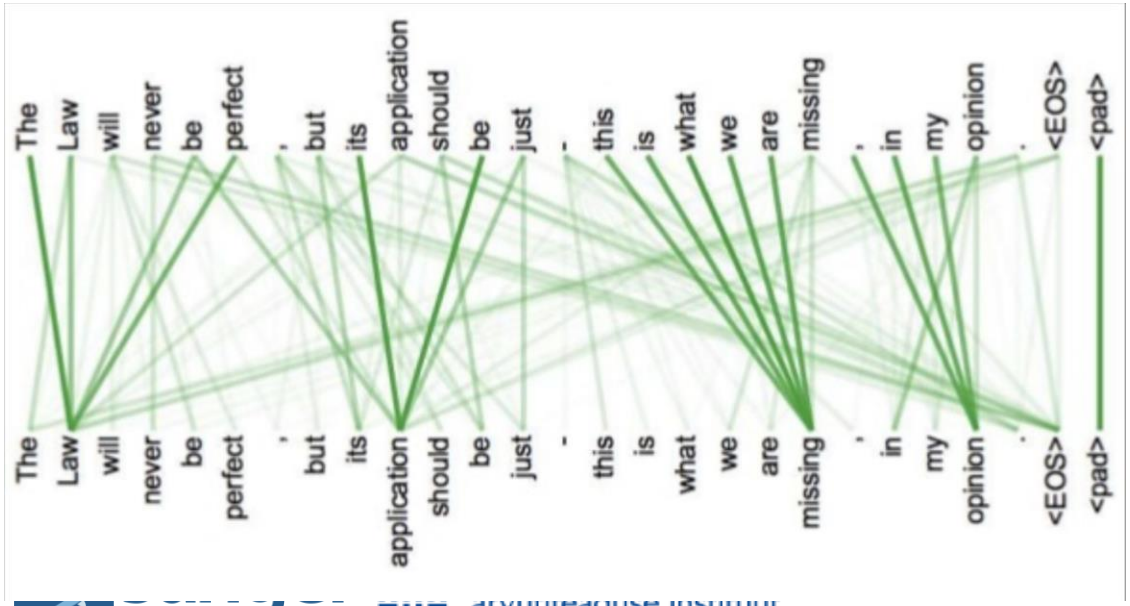


wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for DNA base calling (nanopore)

Attention (idea in LSTM & transformers) – using signal from a distance; can think of as generalization of covariance of hidden states in encoder and decoder.
a_{ij} = weight that decoder for position *i* puts on encoder value at position *j*



Input: current **x**
Target: sequence **y** (e.g. ACCATTGAAC)
Representation

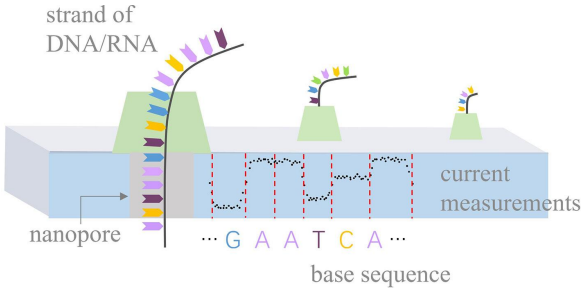
- 5-mer inside pore

Loss function

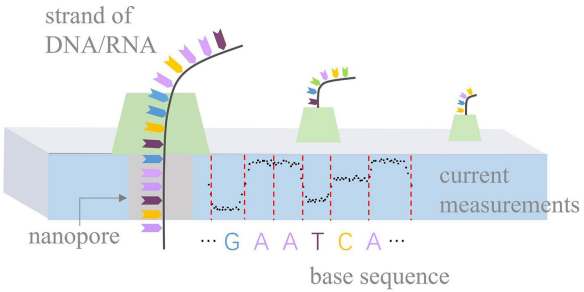
- Base calling error

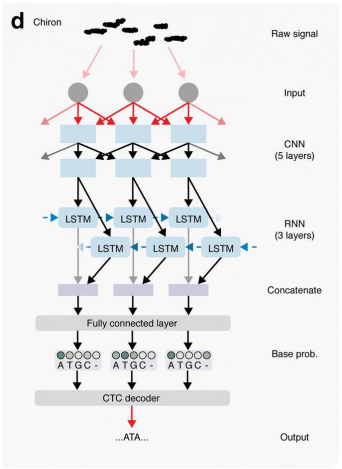
Covariance function

- Internal: linking currents across distance



Deep learning for DNA base calling (nanopore)





Key insight: networks parametrize short- and long-range dependencies in data, relaxing (giving more descriptive power to) natural probabilistic models. The longer scale the dependency (transformers, LSTM), the more data are needed

Example papers:

From squiggle to basepair: computational approaches for improving nanopore sequencing read accuracy. Rang *et al.*, BMC Bioinf 2018

Input: current **x**

Target: sequence **y** (e.g. ACCATTGAAC)

Representation

- 5-mer inside pore

Loss function

- Base calling error

Covariance function

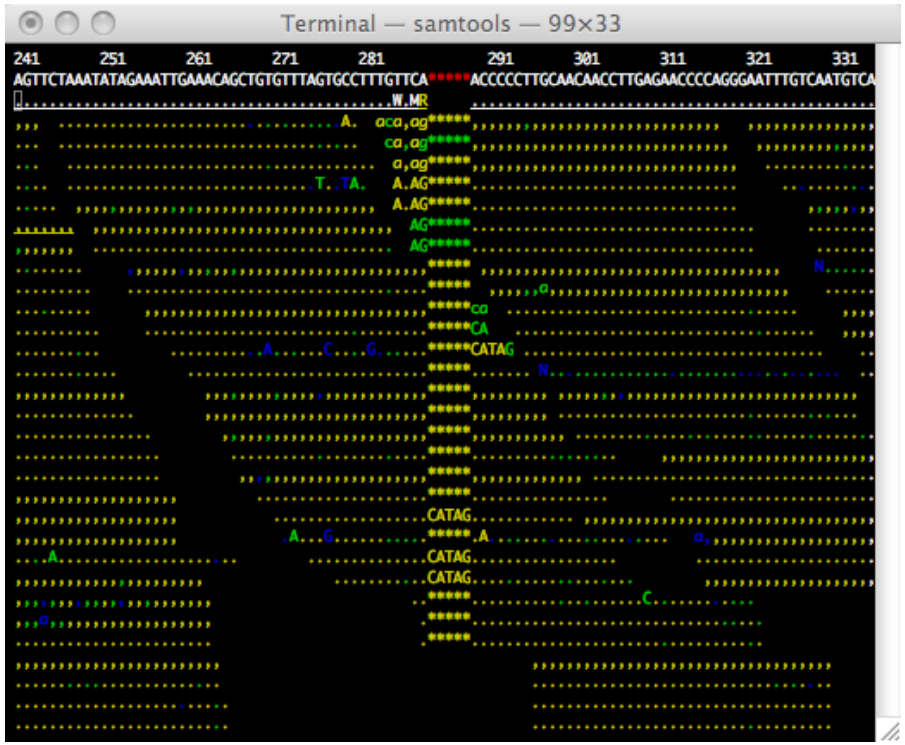
- Internal: linking currents across distance

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for DNA variant calling

x



Input: read pileup **x**

Target: site genotype **y**

Representation?

Loss function?

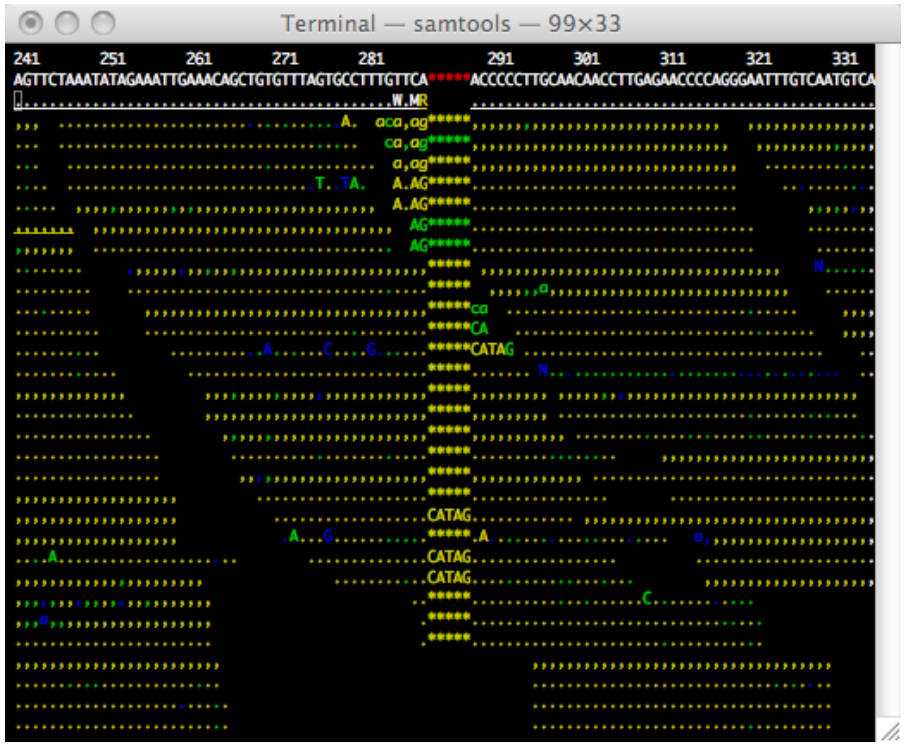
Covariance function?

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for DNA variant calling

x



Input: read pileup **x**

Target: site genotype **y**

Representation

- Count and quality of mapped reference and alternative alleles; other summaries

Loss function?

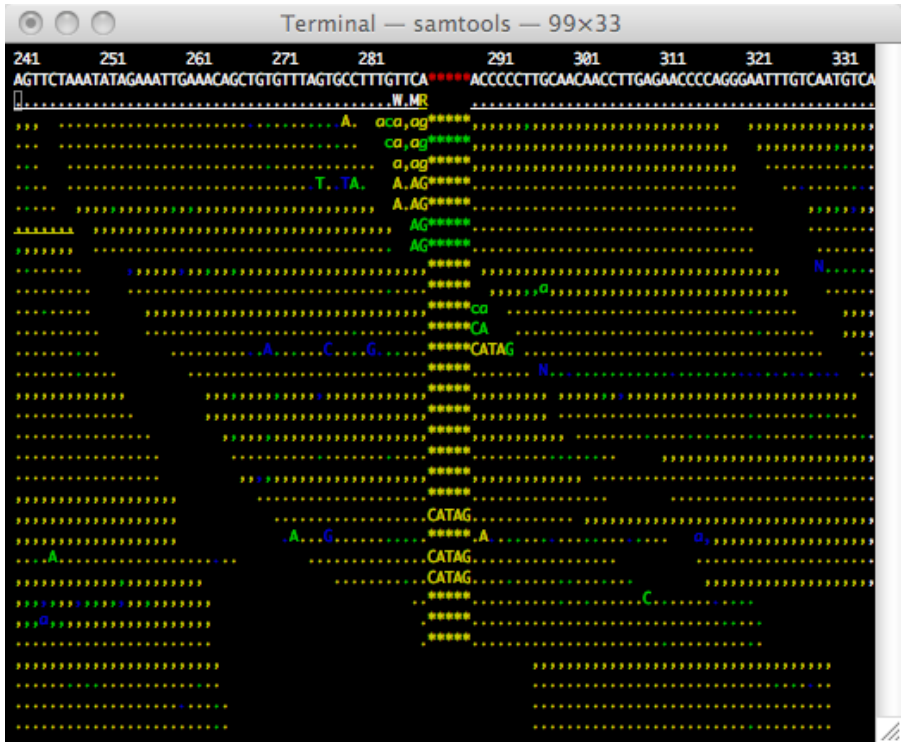
Covariance function?

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for DNA variant calling

x



Input: read pileup **x**

Target: site genotype **y**

Representation

- Count and quality of mapped reference and alternative alleles; other summaries

Loss function

- Genotype accuracy
- A desired mix of precision and recall

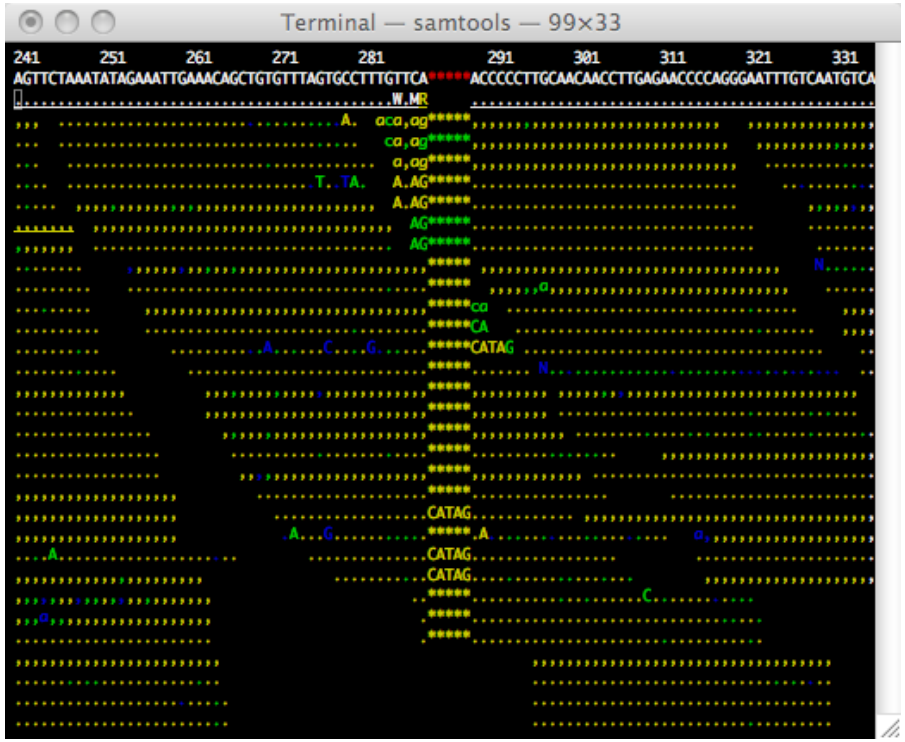
Covariance function?

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for DNA variant calling

Natural model: aggregating signal from reads using Bayes



Input: read pileup **x**

Target: site genotype **y**

Representation

- Count and quality of mapped reference and alternative alleles; other summaries

Loss function

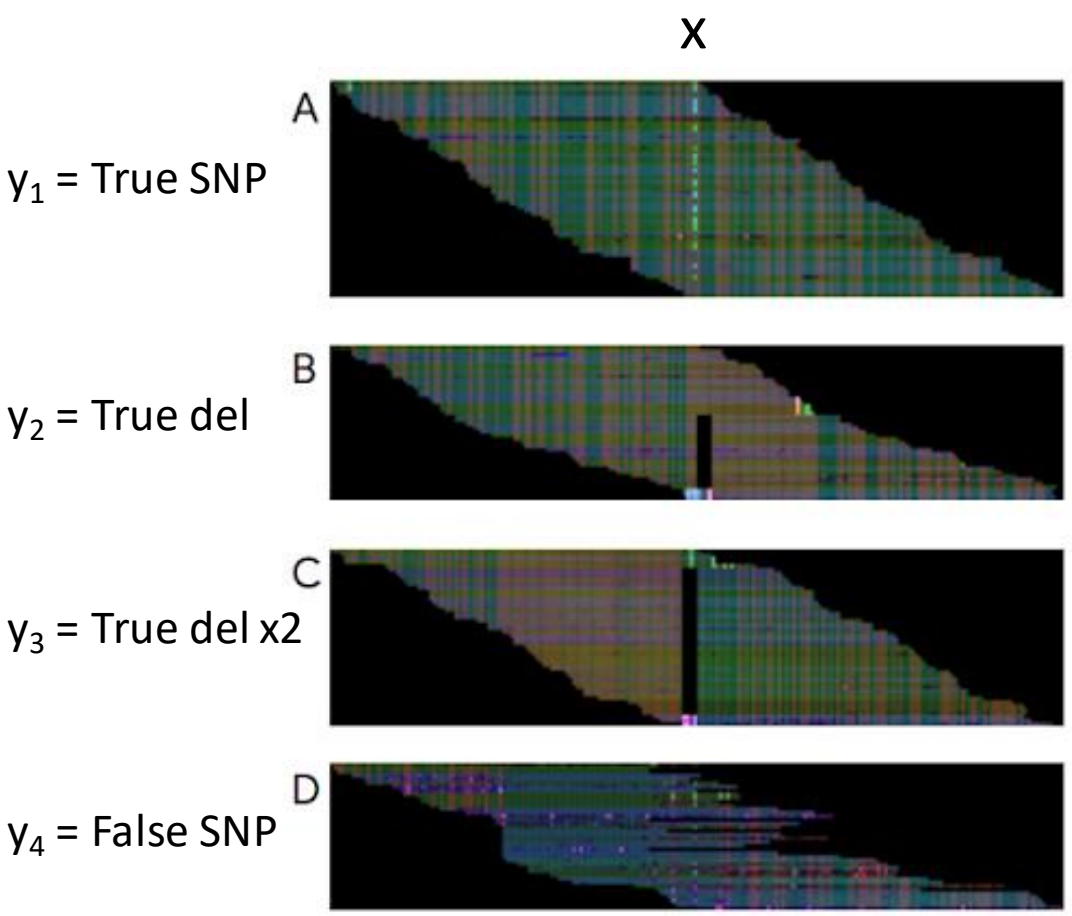
- Genotype accuracy
- A desired mix of precision and recall

Covariance function?

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for DNA variant calling



Input: read pileup **x**

Target: site genotype **y**

Representation

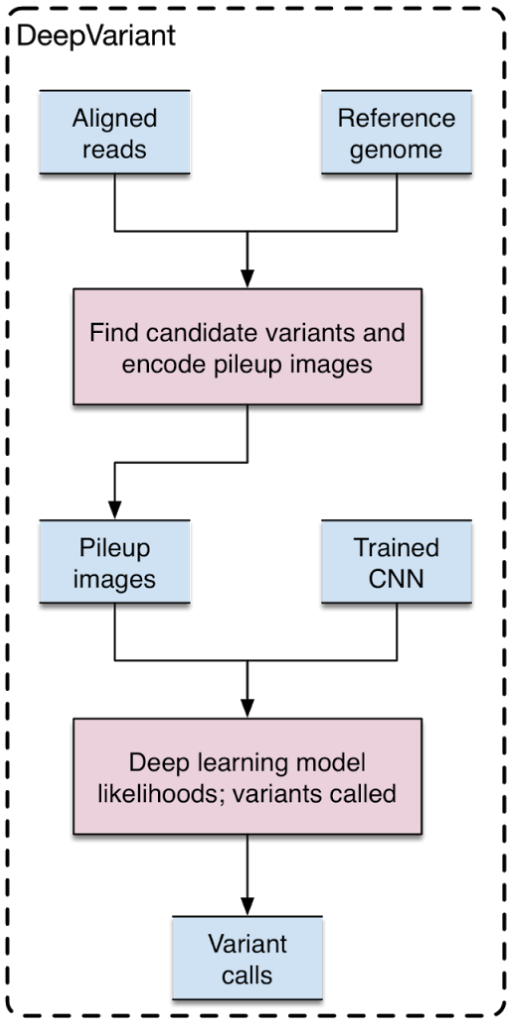
- Count and quality of mapped reference and alternative alleles; other summaries

Loss function

- Genotype accuracy
- A desired mix of precision and recall

Covariance function?

Deep learning for DNA variant calling



Input: read pileup **x**

Target: site genotype **y**

Representation

- Count and quality of mapped reference and alternative alleles; other summaries

Loss function

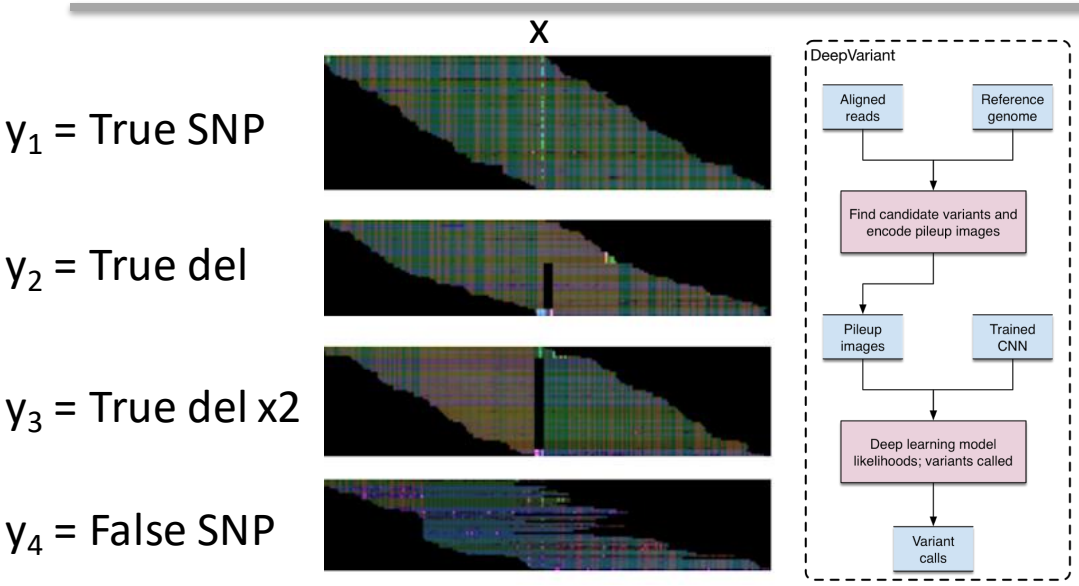
- Genotype accuracy
- A desired mix of precision and recall

Covariance function?

wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

Deep learning for DNA variant calling



Key insight: network operates in the same (visual) space as humans who evaluate, and considers more context than seems necessary at first

Example papers:
A universal SNP and small-indel variant caller using deep neural networks. Poplin et al., NBT 2018

Input: read pileup **x**

Target: site genotype **y**

Representation

- Count and quality of mapped reference and alternative alleles; other summaries

Loss function

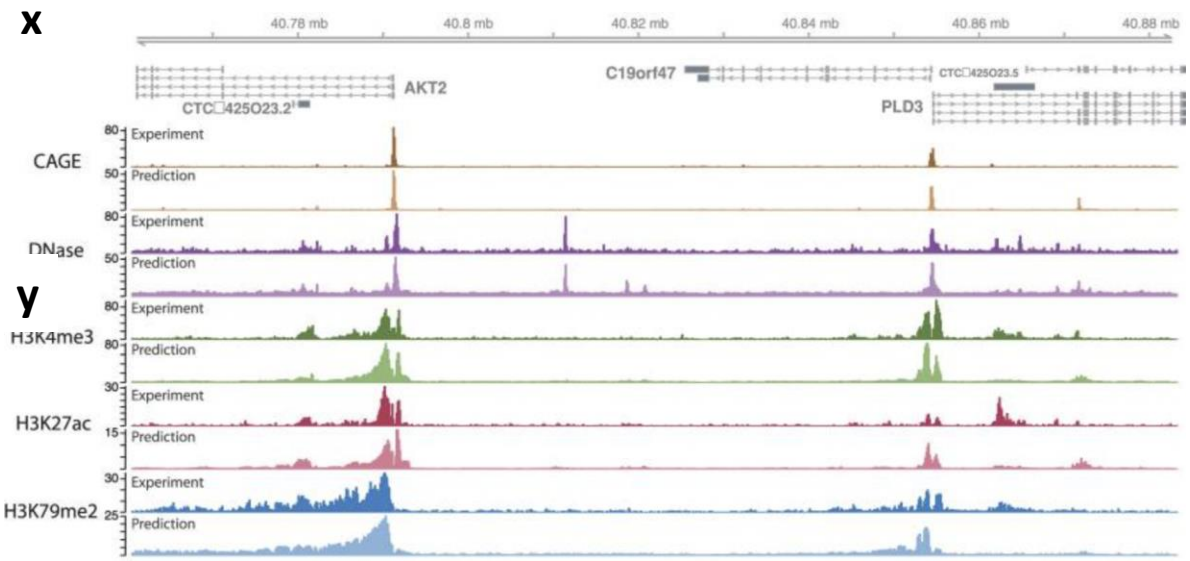
- Genotype accuracy
- A desired mix of precision and recall

Covariance function?

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for gene regulation



Input: sequence **x**

Target: vector of properties **y** (e.g. CTCF binds, H3K9 methylation)

Representation?

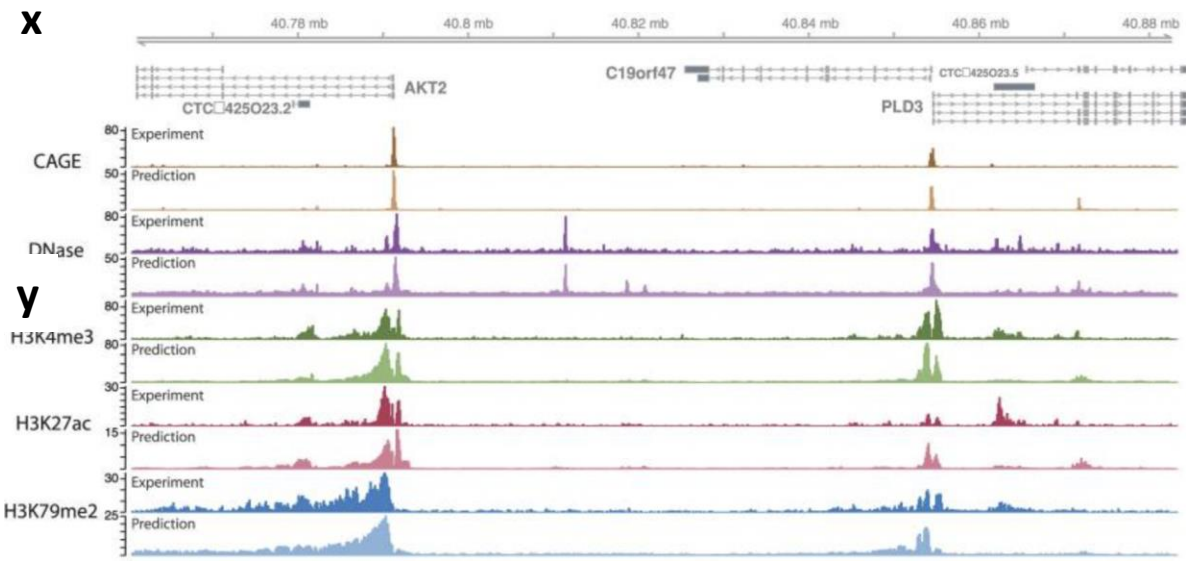
Loss function?

Covariance function?

wellcome trust sanger institute

TARTU ÜLIKOOL arvutiteaduse instituut 1632

Deep learning for gene regulation



Input: sequence **x**

Target: vector of properties **y** (e.g. CTCF binds, H3K9 methylation)

Representation

- Broad chromatin state (active, silenced, …)
- Sequence motifs for protein binding

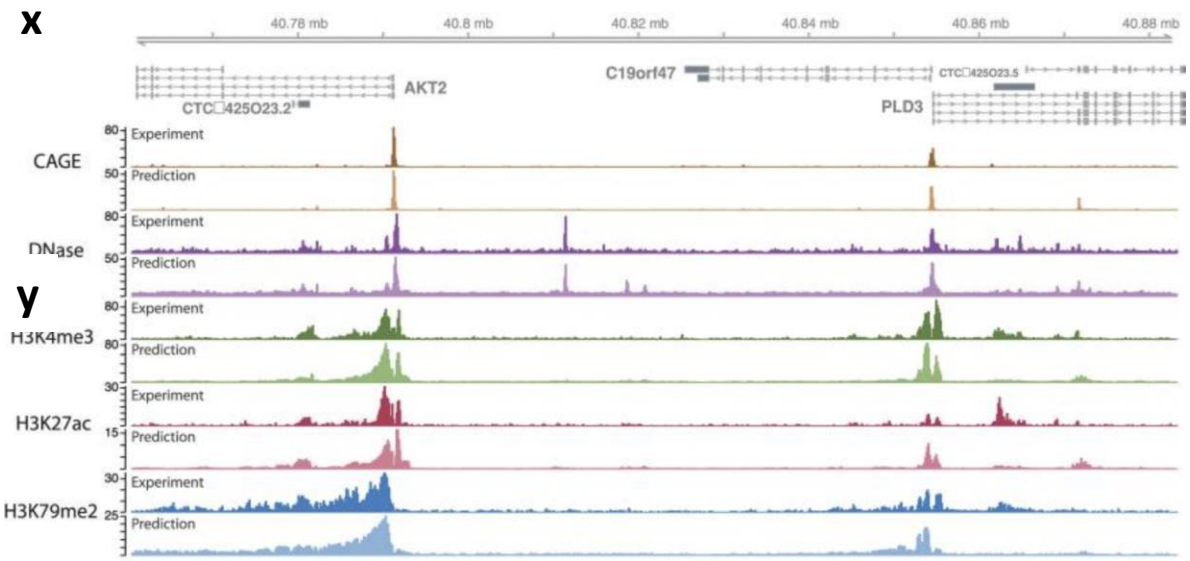
Loss function?

Covariance function?

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for gene regulation



Input: sequence **x**

Target: vector of properties **y** (e.g. CTCF binds, H3K9 methylation)

Representation

- Broad chromatin state (active, silenced, …)
- Sequence motifs for protein binding

Loss function

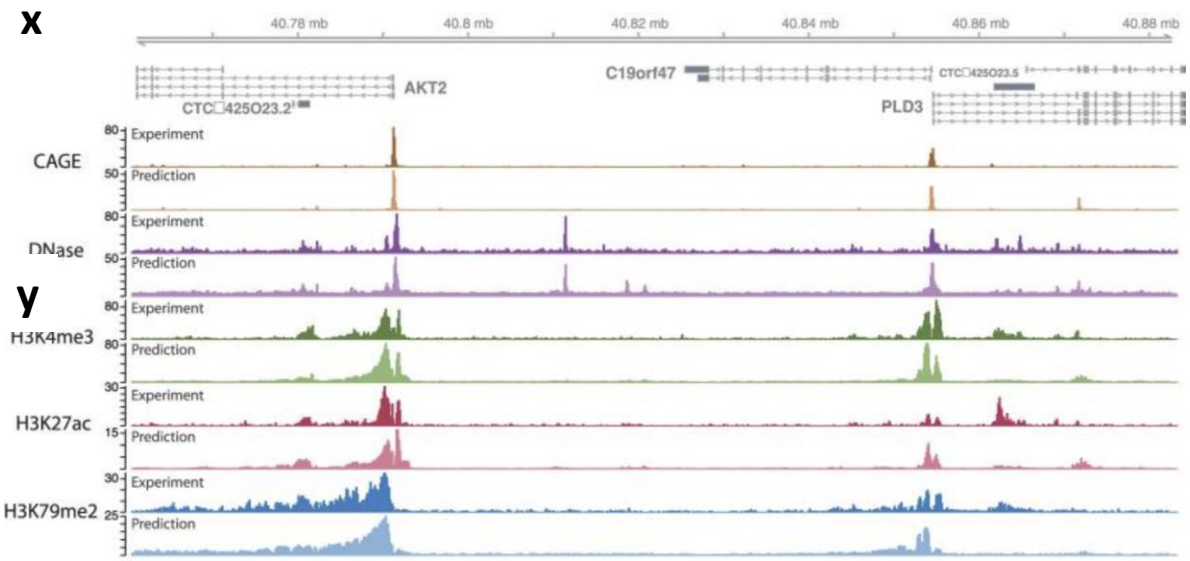
- MSE of signal tracks
- Peak recovery accuracy

Covariance function?

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for gene regulation



Input: sequence **x**

Target: vector of properties **y** (e.g. CTCF binds, H3K9 methylation)

Representation

- Broad chromatin state (active, silenced, …)
- Sequence motifs for protein binding

Loss function

- MSE of signal tracks
- Peak recovery accuracy

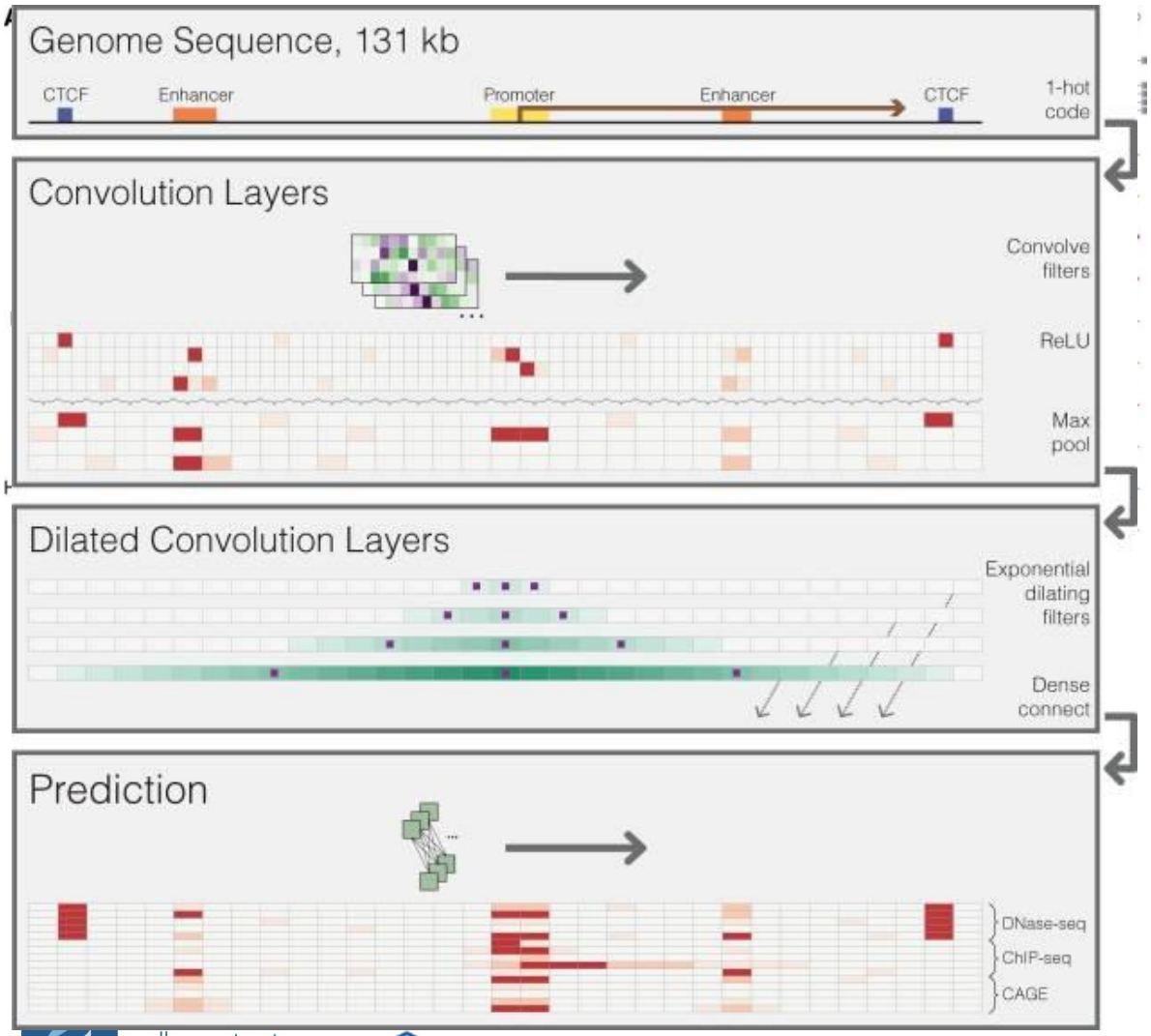
Covariance function

- Similar activity and motif composition

wellcome trust sanger institute

TARTU ÜLIKOOL arvutiteaduse instituut 1632

Deep learning for gene regulation



Input: sequence **x**

Target: vector of properties **y** (e.g. CTCF binds, H3K9 methylation)

Representation

- Broad chromatin state (active, silenced, …)
- Sequence motifs for protein binding

Loss function

- MSE of signal tracks
- Peak recovery accuracy

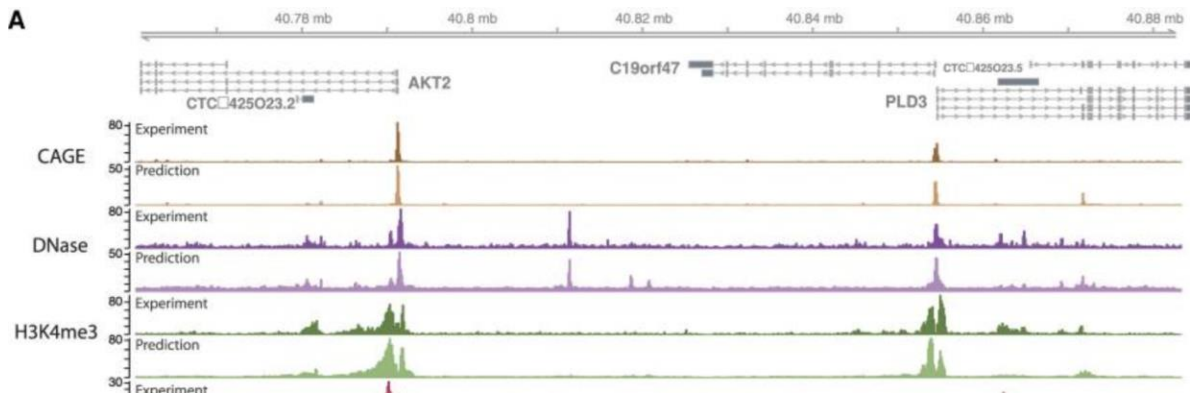
Covariance function

- Similar activity and motif composition

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for gene regulation



Key insights: A single instance of genome can be re-used as many independent inputs (but is too short). If effects are at distance, good models have to embody this, here via dilation, elsewhere e.g. with attention at distance with transformers

Example papers: Effective gene expression prediction from sequence by integrating long-range interactions. Avsec et al., NMeth 2021.

Sequential regulatory activity prediction across chromosomes with convolutional neural networks. Kelley et al., Genome Res 2018

Input: sequence **x**

Target: vector of properties **y** (e.g. CTCF binds, H3K9 methylation)

Representation

- Broad chromatin state (active, silenced, …)
- Sequence motifs for protein binding

Loss function

- MSE of signal tracks
- Peak recovery accuracy

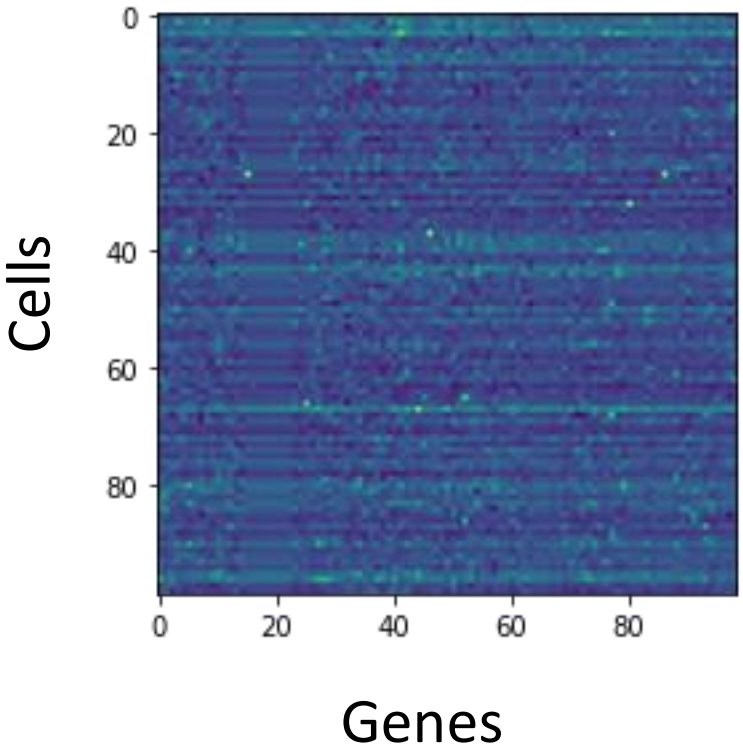
Covariance function

- Similar activity and motif composition

wellcome trust sanger institute

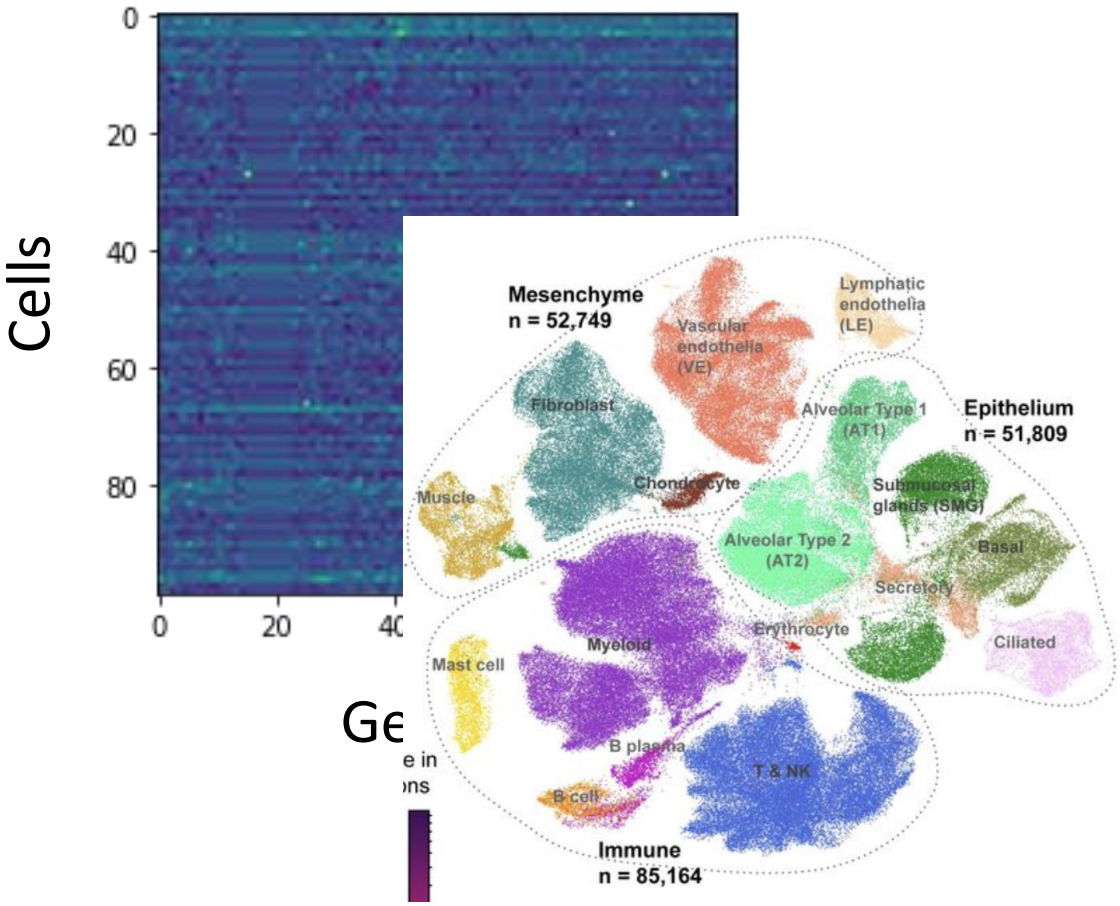
TARTU ÜLIKOOL arvutiteaduse instituut 1632

Deep learning in single cell RNA sequencing



Input: read counts **x**
Target: various (self; cell type; response; time)
Representation?
Loss function?
Covariance function?

Deep learning in single cell RNA sequencing



Input: read counts **x**

Target: various (self; cell type; response; time)

Representation

- Marker expression
- Cell type
- Cluster membership
- Cell state G1/S/G2/M

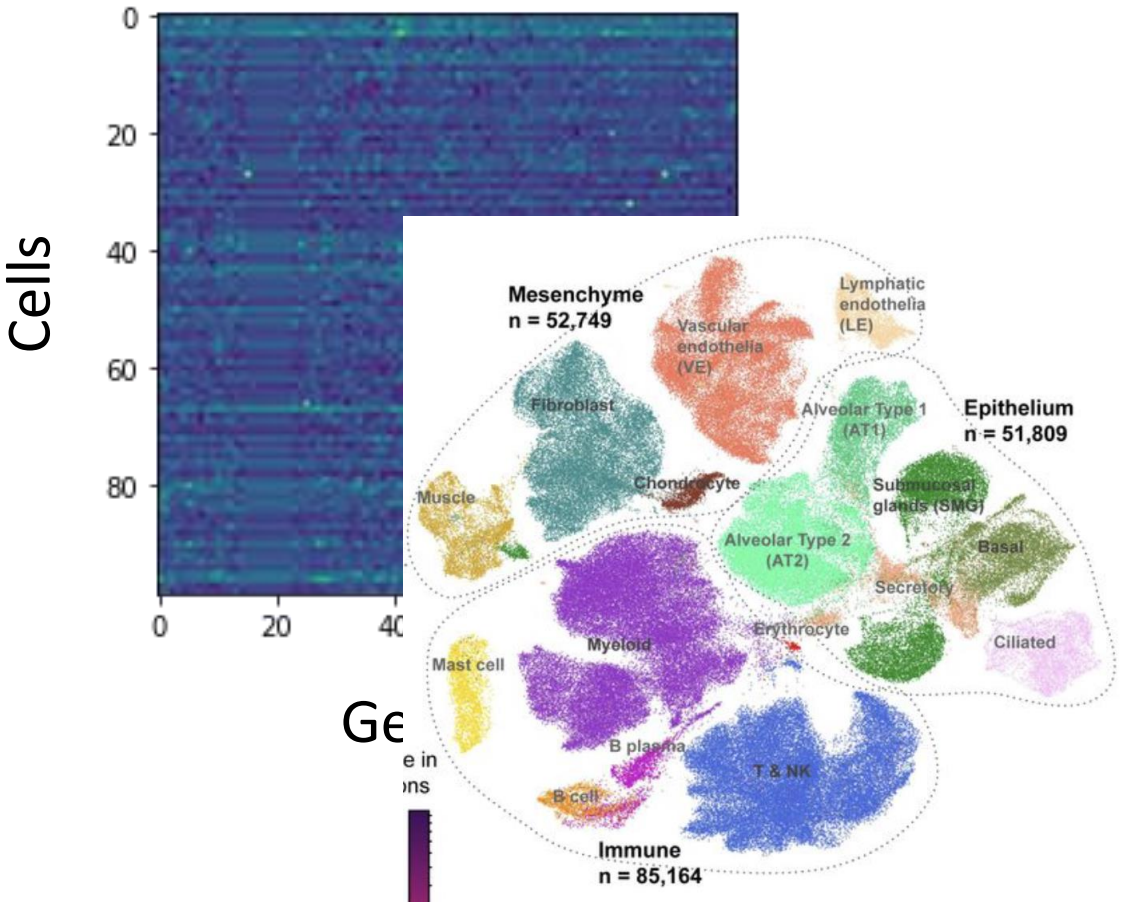
Loss function?

Covariance function?

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in single cell RNA sequencing



Input: read counts **x**

Target: various (self; cell type; response; time)

Representation

- Marker expression
- Cell type
- Cluster membership
- Cell state G1/S/G2/M

Loss function

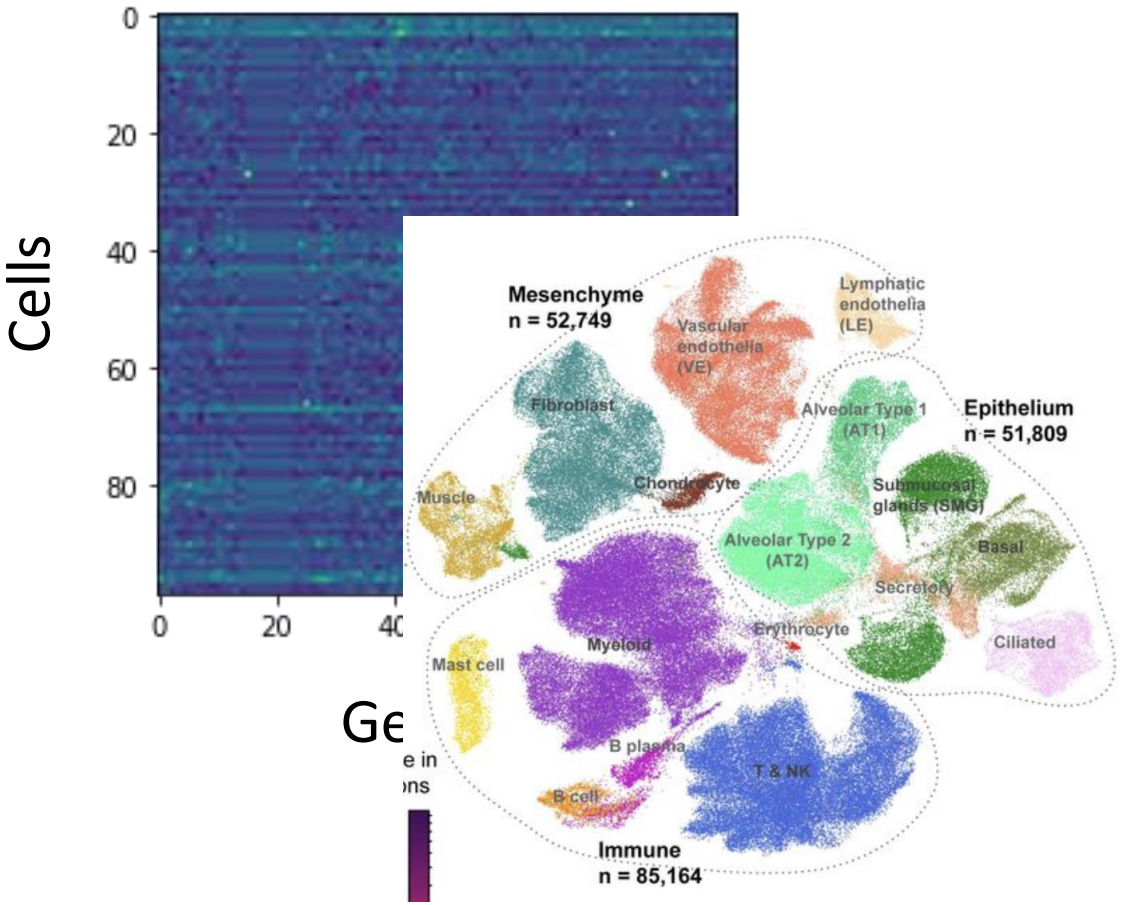
- Mean squared error
- Maxent of residuals

Covariance function?

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in single cell RNA sequencing



Input: read counts **x**

Target: various (self; cell type; response; time)

Representation

- Marker expression
- Cell type
- Cluster membership
- Cell state G1/S/G2/M

Loss function

- Mean squared error
- Maxent of residuals

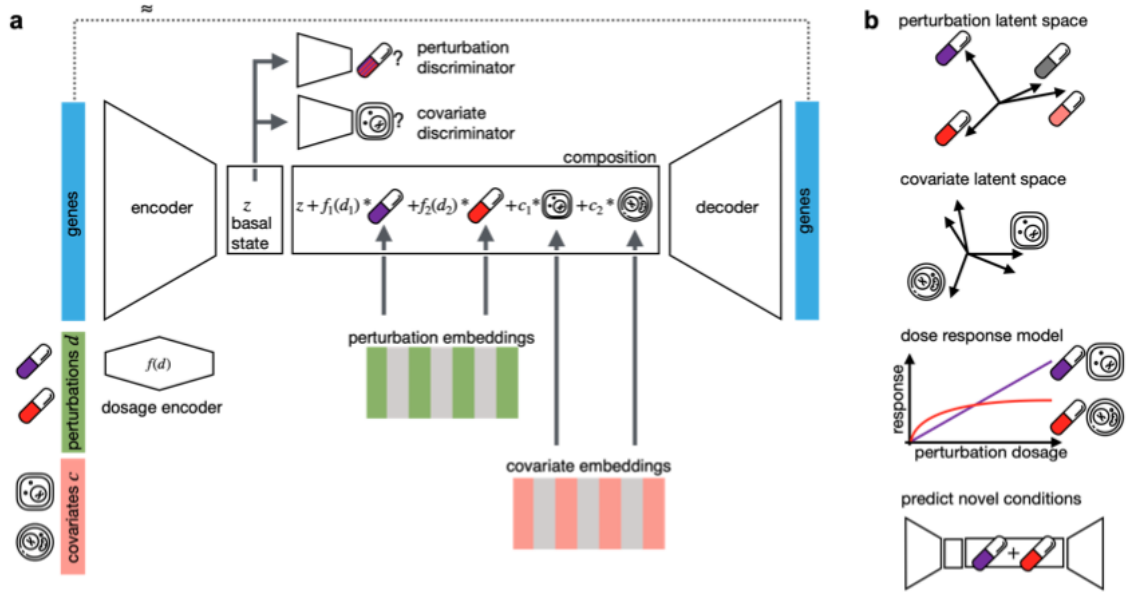
Covariance function

- Correlation
- Co-clustering
- PCA coordinates
- Cosine similarity

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in single cell RNA sequencing



Input: read counts **x**

Target: various (self; cell type; response; time)

Representation

- Marker expression
- Cell type
- Cluster membership
- Cell state G1/S/G2/M

Loss function

- Mean squared error
- Maxent of residuals

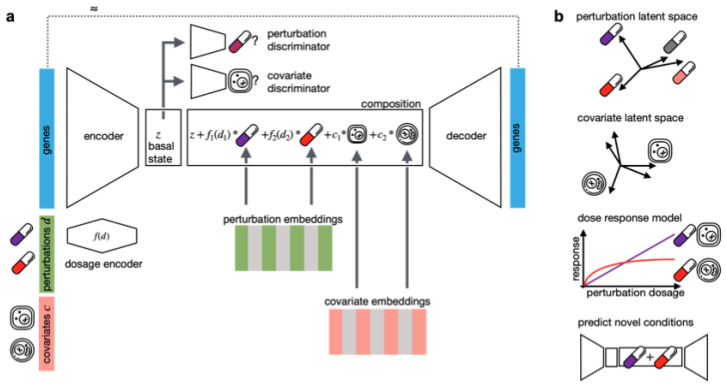
Covariance function

- Correlation
- Co-clustering
- PCA coordinates
- Cosine similarity

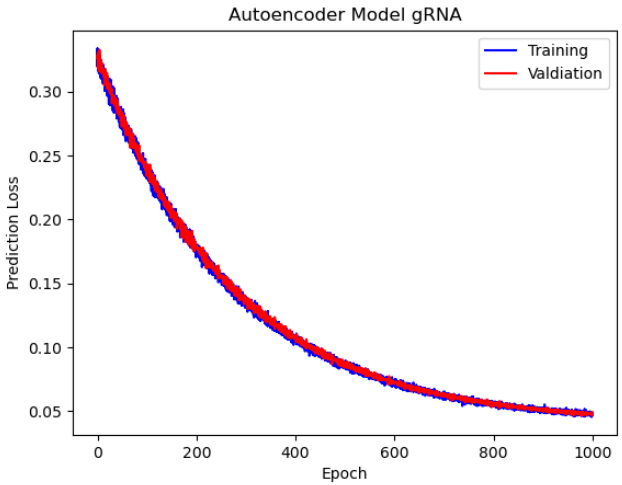
<u>Combinatorial perturbation autoencoder:</u>
Unsupervised learning, but integrating known factors of perturbation state and cell state in the bottleneck layer in adversarial way, in principle allowing extrapolating to new covariate states and perturbations to predict their effect

wellcome trust sanger institute
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in single cell RNA sequencing



Combinatorial perturbation autoencoder: implementation in pytorch



Input: read counts **x**

Target: various (self; cell type; response; time)

Representation

- Marker expression
- Cell type
- Cluster membership
- Cell state G1/S/G2/M

Loss function

- Mean squared error
- Maxent of residuals

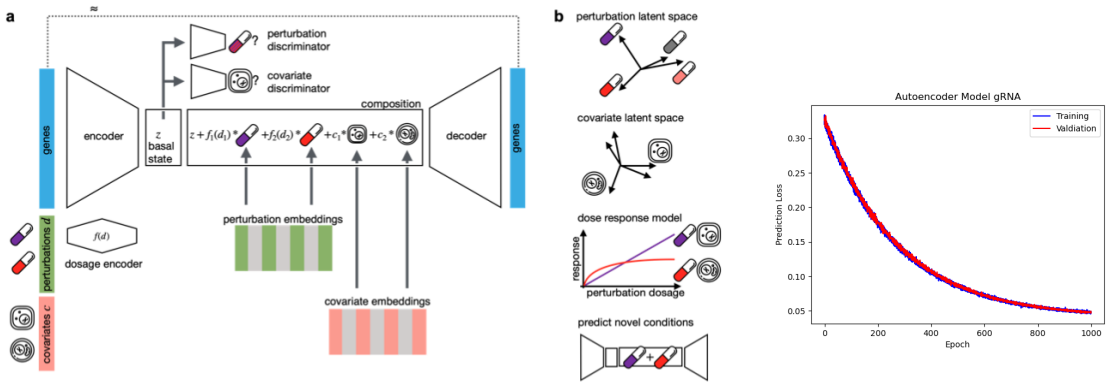
Covariance function

- Correlation
- Co-clustering
- PCA coordinates
- Cosine similarity

wellcome trust sanger institute

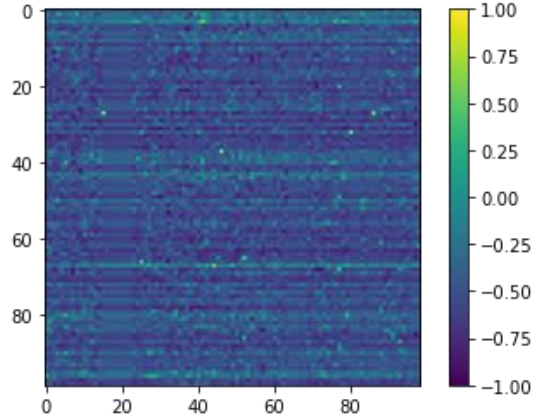
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in single cell RNA sequencing

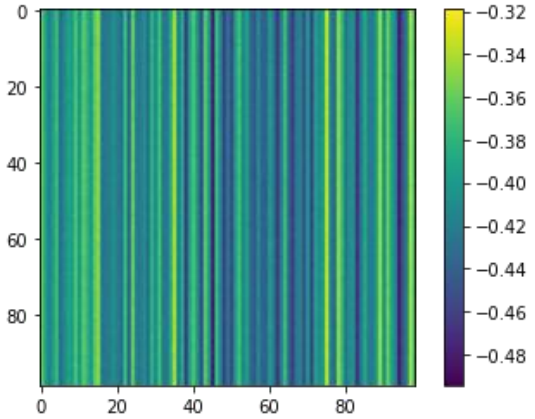


Combinatorial perturbation autoencoder:
implementation in pytorch; beware dragons

True gene expression



Predicted gene expression



Input: read counts **x**

Target: various (self; cell type; response; time)

Representation

- Marker expression
- Cell type
- Cluster membership
- Cell state G1/S/G2/M

Loss function

- Mean squared error
- Maxent of residuals

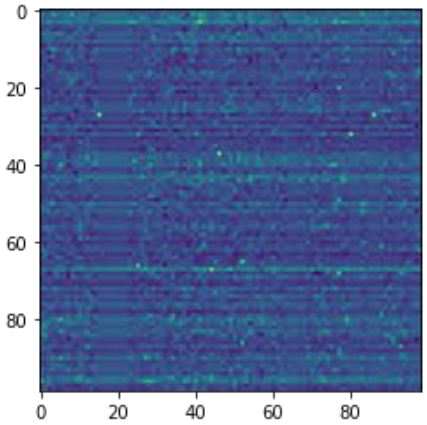
Covariance function

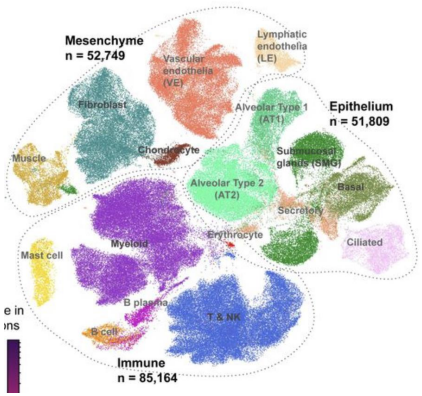
- Correlation
- Co-clustering
- PCA coordinates
- Cosine similarity

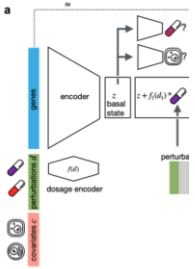
wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in single cell RNA sequencing







Key insight: given huge datasets, most models amount to compression, and advances are not trivial to evaluate for benefit. Choices of normalization and latent space representation make a substantial difference to outputs.

Example papers: Deep generative modeling for single-cell transcriptomics. Lopez et al. NMeth, 2018
Learning interpretable cellular responses to complex perturbations in high-throughput screens. Lotfollahi et al. bioRxiv, 2021.

Input: read counts **x**

Target: various (self; cell type; response; time)

Representation

- Marker expression
- Cell type
- Cluster membership
- Cell state G1/S/G2/M

Loss function

- Mean squared error
- Maxent of residuals

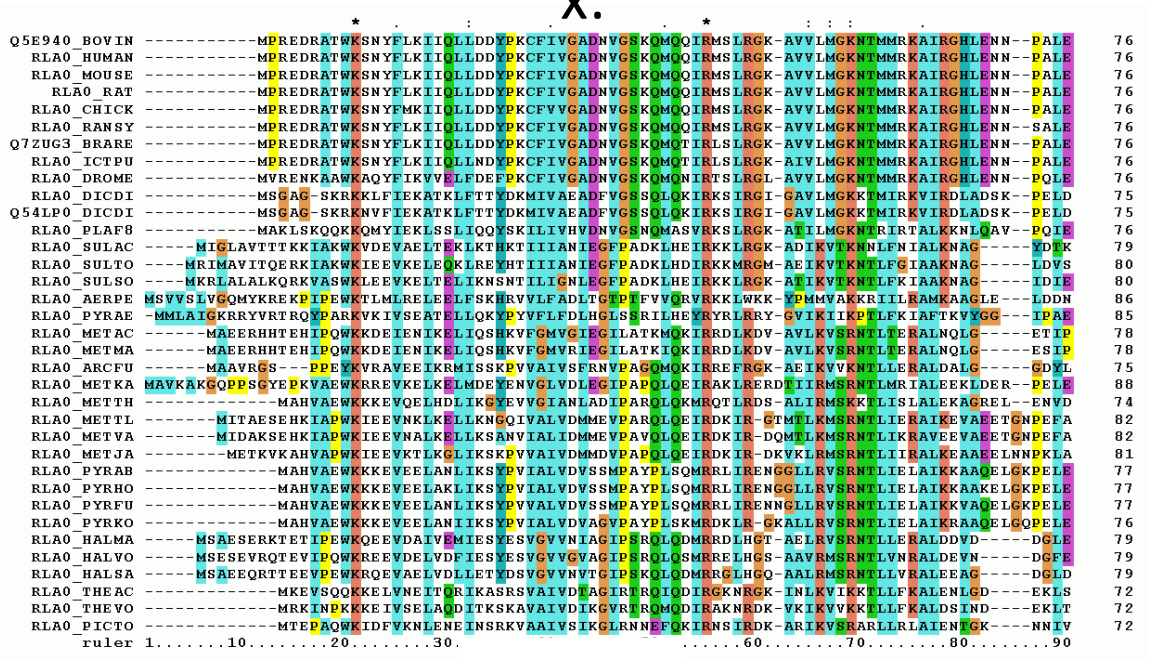
Covariance function

- Correlation
- Co-clustering
- PCA coordinates
- Cosine similarity

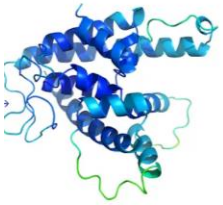
wellcome trust sanger institute
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in protein folding

x:



y:



3D structure

Input: multiple sequence alignment **x**

Target: 3D structure **y**

Representation?

Loss function?

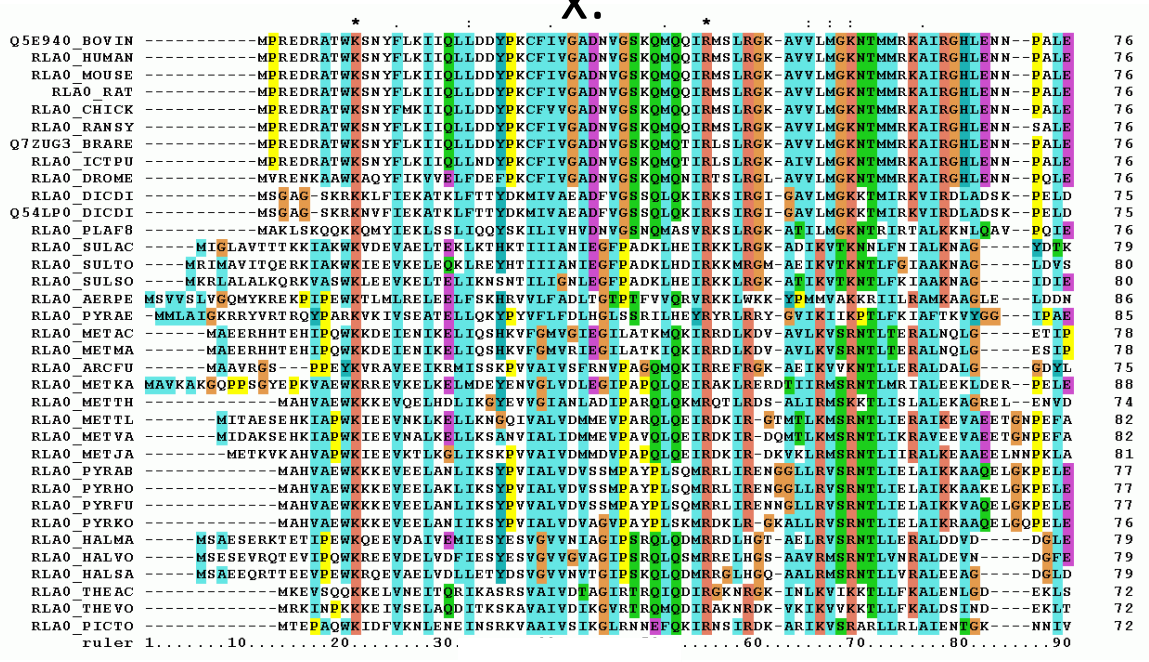
Covariance function?

wellcome trust sanger institute

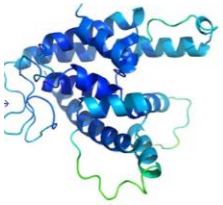
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in protein folding

x:



y:



3D structure

Input: multiple sequence alignment **x**

Target: 3D structure **y**

Representation

- Evolutionary coupling
- Amino acid distances
- Angles in peptide

Loss function?

Covariance function?

wellcome trust sanger institute

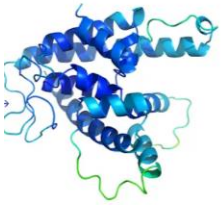
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in protein folding

x:

```
                                  *   .    :    .    .   *      : : :     .
Q5E940_BOVIN -----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQQIRMSLRGK-AVVLMGKNTMMRKAIRGHLENN--PALE   76
  RLA0_HUMAN -----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQQIRMSLRGK-AVVLMGKNTMMRKAIRGHLENN--PALE   76
  RLA0_MOUSE -----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQQIRMSLRGK-AVVLMGKNTMMRKAIRGHLENN--PALE   76
    RLA0_RAT -----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQQIRMSLRGK-AVVLMGKNTMMRKAIRGHLENN--PALE   76
  RLA0_CHICK -----------MPREDRATWKSNYFMKIIQLLDDYPKCFVVGADNVGSKQMQQIRMSLRGK-AVVLMGKNTMMRKAIRGHLENN--PALE   76
  RLA0_RANSY -----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQQIRMSLRGK-AVVLMGKNTMMRKAIRGHLENN--SALE   76
Q7ZUG3_BRARE -----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQTIRLSLRGK-AVVLMGKNTMMRKAIRGHLENN--PALE   76
  RLA0_ICTPU -----------MPREDRATWKSNYFLKIIQLLNDYPKCFIVGADNVGSKQMQTIRLSLRGK-AIVLMGKNTMMRKAIRGHLENN--PALE   76
  RLA0_DROME -----------MVRENKAAWKAQYFIKVVELFDEFPKCFIVGADNVGSKQMQNIRTSLRGL-AVVLMGKNTMMRKAIRGHLENN--PQLE   76
  RLA0_DICDI -----------MSGAG-SKRKKLFIEKATKLFTTYDKMIVAEADFVGSSQLQKIRKSIRGI-GAVLMGKKIMIRKVIRDLADSK--PELD   75
Q54LP0_DICDI -----------MSGAG-SKRKNVFIEKATKLFTTYDKMIVAEADFVGSSQLQKIRKSIRGI-GAVLMGKKIMIRKVIRDLADSK--PELD   75
  RLA0_PLAF8 -----------MAKLSKQQKKQMYIEKLSSLIQQYSKILIVHVDNVGSNQMASVRKSLRGK-ATILMGKNTRIRTALKKNLQAV--PQIE   76
  RLA0_SULAC -----MIGLAVTTTKKIAKWKVDEVAELTEKLKTHKTIIIANIEGFPADKLHEIRKKLRGK-ADIKVTKNNLFNIALKNAG-----YDIK   79
  RLA0_SULTO ----MRIMAVITQERKIAKWKIEEVKELEQKLREYHTIIIANIEGFPADKLHDIRKKMRGM-AEIKVTKNTLFGIAAKNAG-----LDVS   80
  RLA0_SULSO ----MKRLALALKQRKVASWKLEEVKELTELIKNSNTILIGNLEGFPADKLHEIRKKLRGK-ATIKVTKNTLFKIAAKNAG-----IDIE   80
  RLA0_AERPE MSVVSLVGQMYKREKPIPEWKTLMLRELEELFSKHRVVLFADLTGTPTFVVQRVRKKLWKK-YPMMVAKKRIILRAMKAAGLE---LDDN   86
  RLA0_PYRAE -MMLAIGKRRYVRTRQYPARKVKIVSEATELLQKYPYYFLFDLHGLSSRILHEYRYRLRRY-GVIKIIKPTLFKIAFTKVYGG---IPAE   85
  RLA0_METAC ------MAEERHHTEHIPQWKKDEIENIKELIQSHKVFGMVGIEGILATKMQKIRRDLKDV-AVLKVSRNTLTERALNQLG-----ETIP   78
  RLA0_METMA ------MAEERHHTEHIPQWKKDEIENIKELIQSHKVFGMVRIEGILATKIQKIRRDLKDV-AVLKVSRNTLTERALNQLG-----ESIP   78
  RLA0_ARCFU ------MAAVRGS---PPEYKVRAVEEIKRMISSKPVVAIVSFRNVPAGQMQKIRREFRGK-AEIKVVKNTLLERALDALG-----GDYL   75
  RLA0_METKA MAVKAKGQPPSGYEPKVAEWKRREVKELKELMDEYENVGLVDLEGIPAPQLQEIRAKLRERDTIIRMSRNTLMRIALEEKLDER--PELE   88
  RLA0_METTH ------------MAHVAEWKKKEVQELHDLIKGYEVVGIANLADIPARQLQKMRQTLRDS-ALIRMSKKTLISLALEKAGREL--ENVD   74
  RLA0_METTL -------MITAESEHKIAPWKIEEVNKLKELLKNGQIVALVDMMEVPARQLQEIRDKIR-GTMTLKMSRNTLIERAIKEVAEETGNPEFA   82
  RLA0_METVA -------MIDAKSEHKIAPWKIEEVNALKELLKSANVIALIDMMEVPAVQLQEIRDKIR-DQMTLKMSRNTLIKRAVEEVAEETGNPEFA   82
  RLA0_METJA --------METKVKAHVAPWKIEEVKTLKGLIKSKPVVAIVDMMDVPAPQLQEIRDKIR-DKVKLRMSRNTLIIRALKEAAEELNNPKLA   81
  RLA0_PYRAB ------------MAHVAEWKKKEVEELANLIKSYPVIALVDVSSMPAYPLSQMRRLIRENGGLLRVSRNTLIELAIKKAAQELGKPELE   77
  RLA0_PYRHO ------------MAHVAEWKKKEVEELAKLIKSYPVIALVDVSSMPAYPLSQMRRLIRENGGLLRVSRNTLIELAIKKAAKELGKPELE   77
  RLA0_PYRFU ------------MAHVAEWKKKEVEELANLIKSYPVVALVDVSSMPAYPLSQMRRLIRENNGLLRVSRNTLIELAIKKVAQELGKPELE   77
  RLA0_PYRKO ------------MAHVAEWKKKEVEELANIIKSYPVIALVDVAGVPAYPLSKMRDKLR-GKALLRVSRNTLIELAIKRAAQELGQPELE   76
  RLA0_HALMA -----MSAESERKTETIPEWKQEEVDAIVEMIESYESVGVVNIAGIPSRQLQDMRRDLHGT-AELRVSRNTLLERALDDVD-----DGLE   79
  RLA0_HALVO -----MSESEVRQTEVIPQWKREEVDELVDFIESYESVGVVGVAGIPSRQLQSMRRELHGS-AAVRMSRNTLVNRALDEVN-----DGFE   79
  RLA0_HALSA -----MSAEEQRTTEEVPEWKRQEVAELVDLLETYDSVGVVNVTGIPSKQLQDMRRGLHGQ-AALRMSRNTLLVRALEEAG-----DGLD   79
  RLA0_THEAC ------------MKEVSQQKKELVNEITQRIKASRSVAIVDTAGIRTRQIQDIRGKNRGK-INLKVIKKTLLFKALENLGD----EKLS   72
  RLA0_THEVO ------------MRKINPKKKEIVSELAQDITKSKAVAIVDIKGVRTRQMQDIRAKNRDK-VKIKVVKKTLLFKALDSIND----EKLT   72
  RLA0_PICTO ------------MTEPAQWKIDFVKNLENEINSRKVAAIVSIKGLRNNEFQKIRNSIRDK-ARIKVSRARLLRLAIENTGK----NNIV   72
       ruler 1.......10........20........30.                  .....60........70........80........90
```

y:



3D structure

Input: multiple sequence alignment **x**

Target: 3D structure **y**

Representation

- Evolutionary coupling
- Amino acid distances
- Angles in peptide

Loss function

- Mean squared error of angles
- Mean squared error of distances

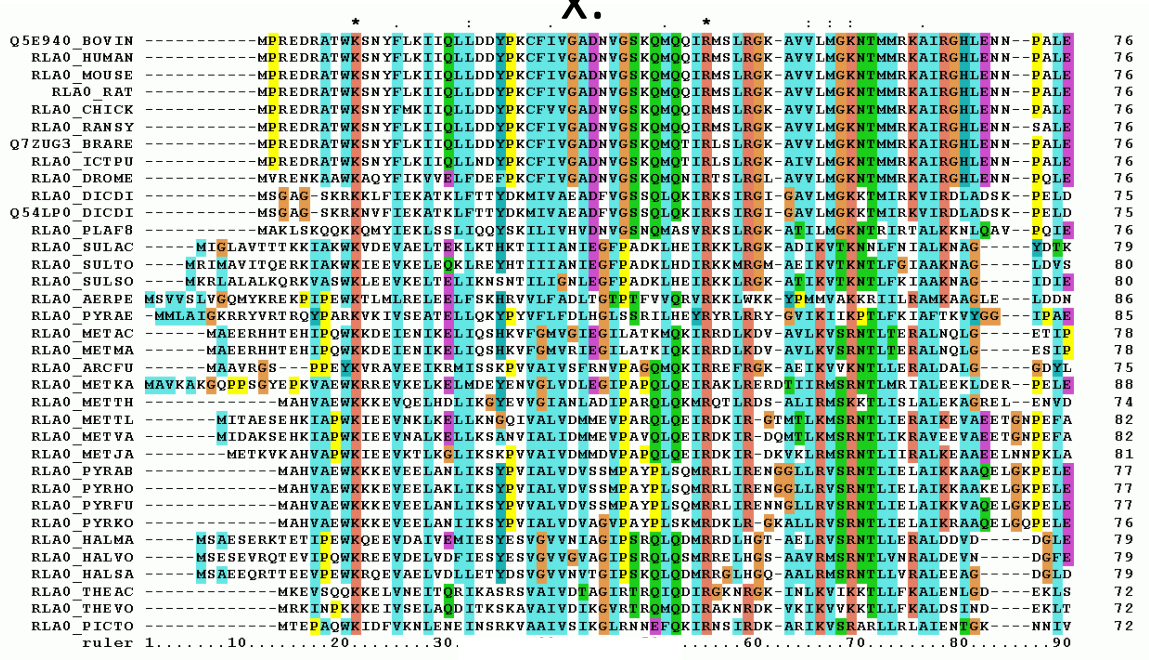
Covariance function?

wellcome trust sanger institute

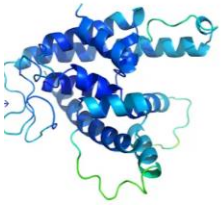
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in protein folding

x:



y:



3D structure

Input: multiple sequence alignment **x**

Target: 3D structure **y**

Representation

- Evolutionary coupling
- Amino acid distances
- Angles in peptide

Loss function

- Mean squared error of angles
- Mean squared error of distances

Covariance function

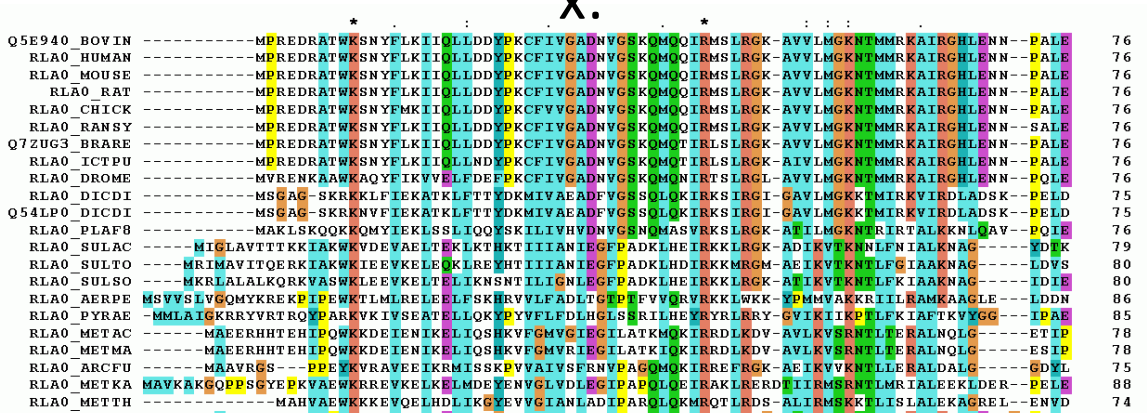
- Sharing structure elements (e.g. helices)

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning in protein folding

x:

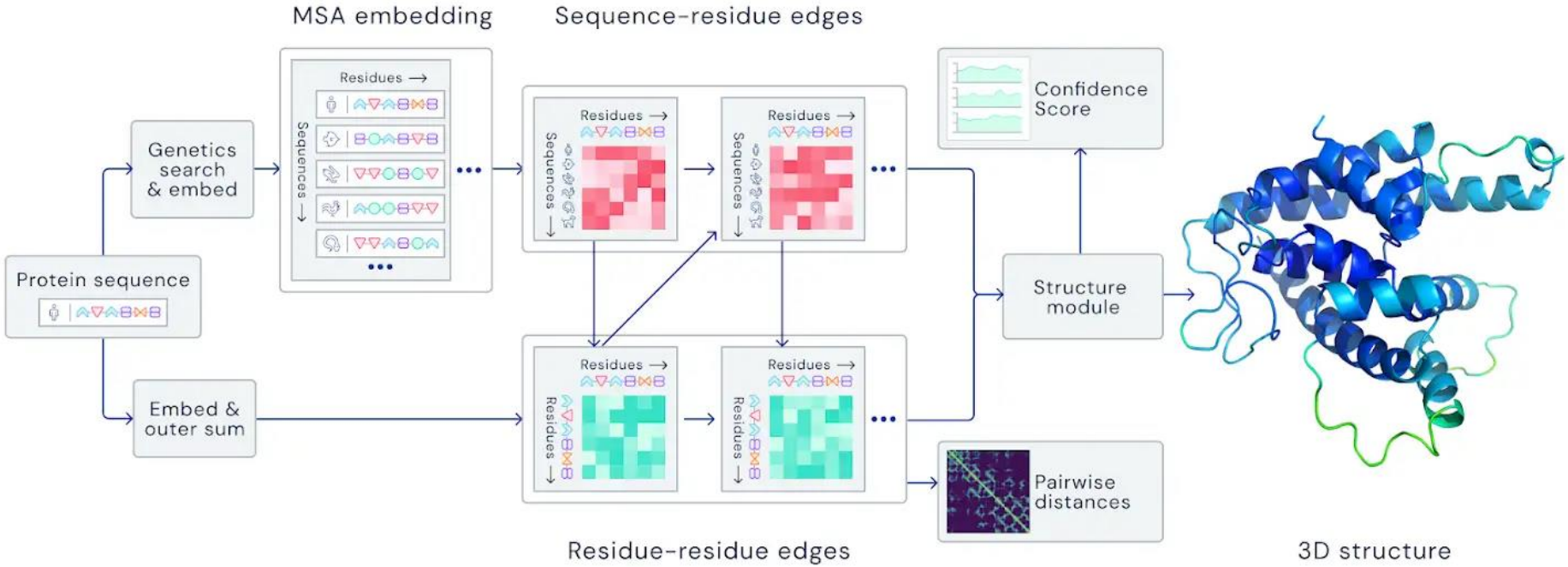


Input: multiple sequence alignment **x**

Target: 3D structure **y**

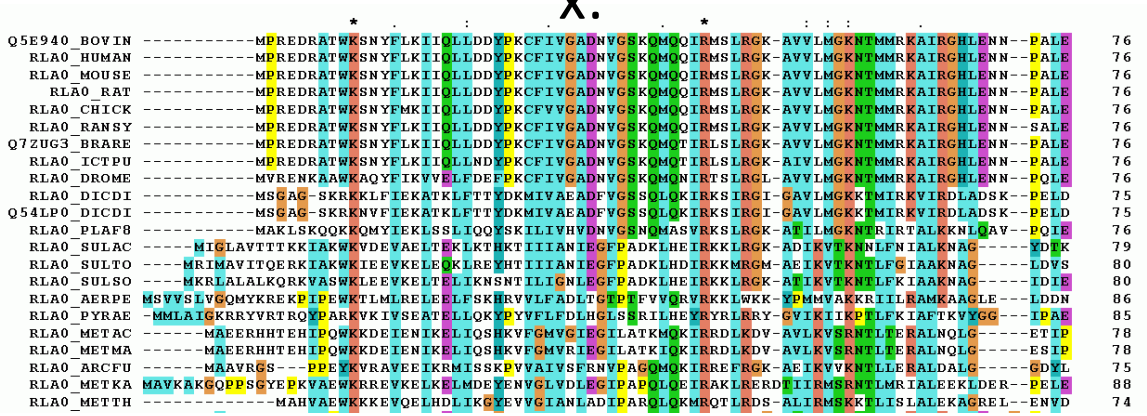
Representation

- Evolutionary coupling
- Amino acid distances



Deep learning in protein folding

x:

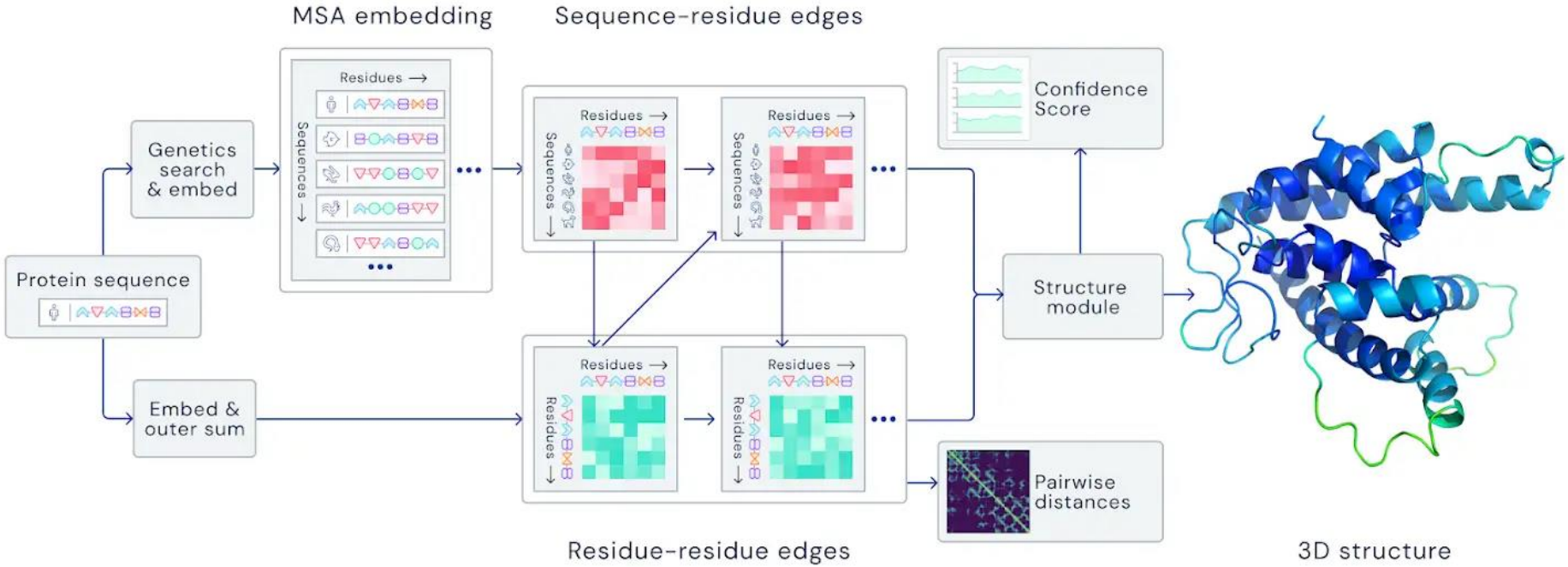


Input: multiple sequence alignment **x**

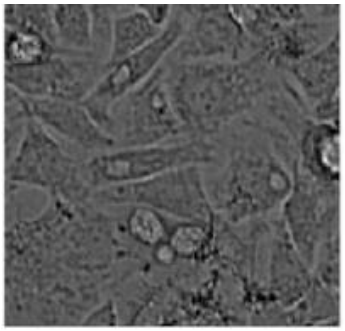
Target: 3D structure **y**

Representation

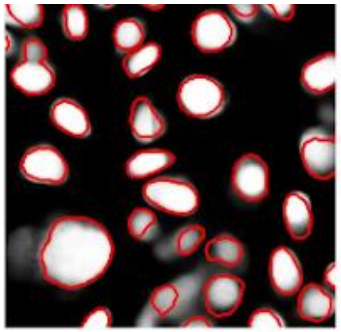
- Evolutionary coupling
- Amino acid distances



Deep learning for cell counting from images



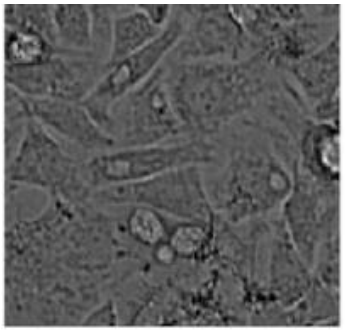
x



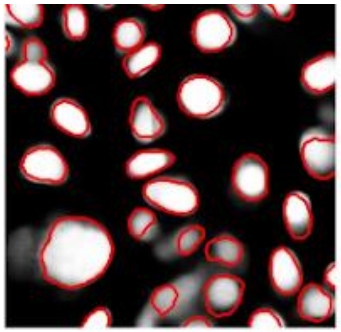
y

Input: brightfield microscopy image **x**
Target: nuclear masks **y**
Representation?
Loss function?
Covariance function?

Deep learning for cell counting from images



x



y

Input: brightfield microscopy image **x**

Target: nuclear masks **y**

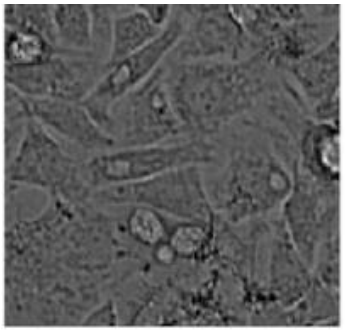
Representation

- Intensity, texture features, …

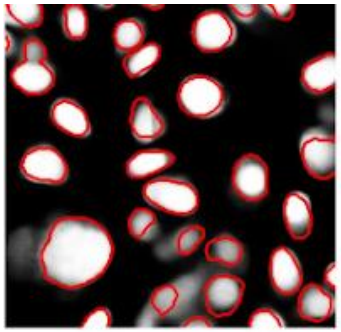
Loss function?

Covariance function?

Deep learning for cell counting from images



x



y

Input: brightfield microscopy image **x**

Target: nuclear masks **y**

Representation

- Intensity, texture features, …

Loss function

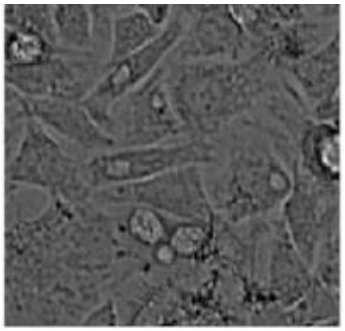
- Mean squared error
- Pixel accuracy
- Object accuracy
- Count accuracy

Covariance function?

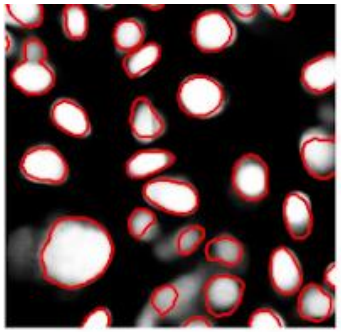
wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for cell counting from images



x



y

Input: brightfield microscopy image **x**

Target: nuclear masks **y**

Representation

- Intensity, texture features, …

Loss function

- Mean squared error
- Pixel accuracy
- Object accuracy
- Count accuracy

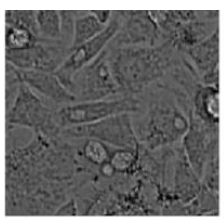
Covariance function

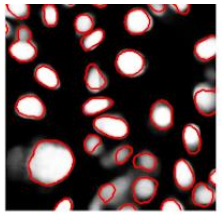
- From features

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

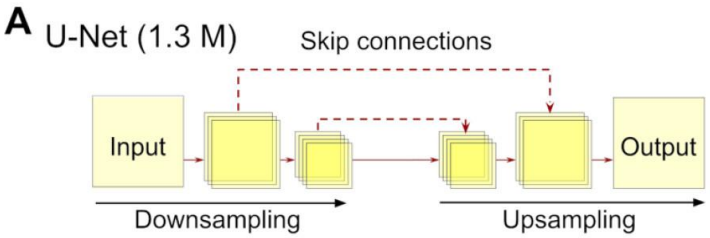
Deep learning for cell counting from images

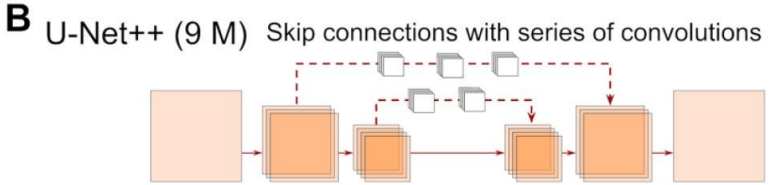


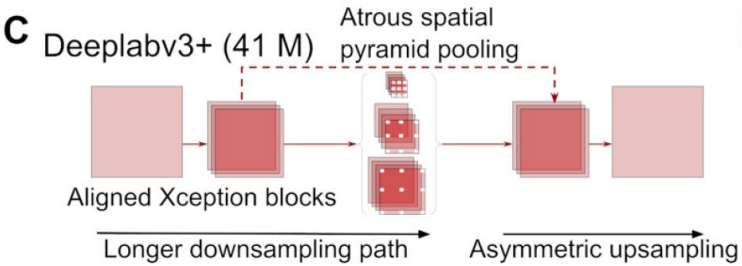


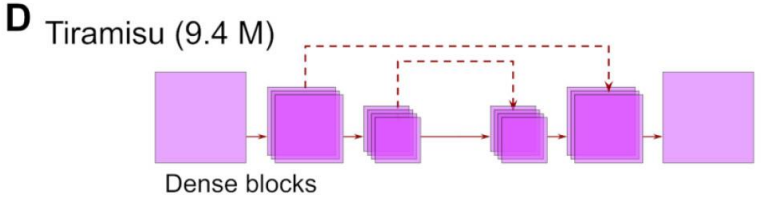
Input: brightfield microscopy image **x**

Target: nuclear masks **y**

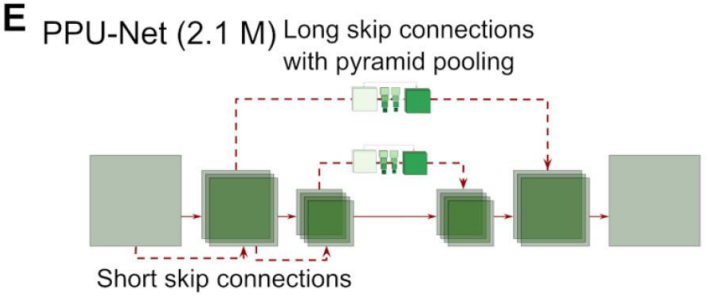


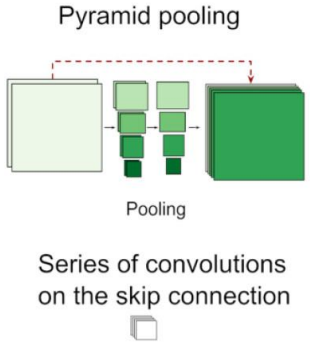






or

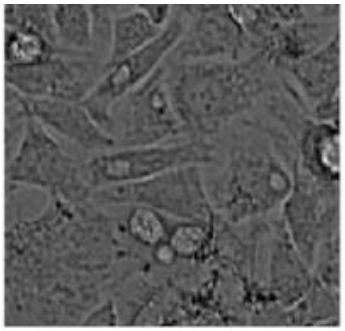




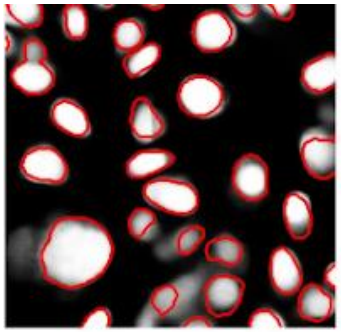
F Average prediction time for 1080x1080 image in seconds

U-Net	0.281
U-Net++	0.333
Deeplabv3+	0.243
Tiramisu	1.424
PPU-Net	0.228

Deep learning for cell counting from images



x



y

Key insight: texture and other local variation give signal that is not easily human-detected, making achieving super-human performance (providing value) easy. Architecture blocks can be reused in plug-and-play manner, but cooler blocks and more parameters do not imply better results.

Example papers: Evaluating Very Deep Convolutional Neural Networks for Nucleus Segmentation from Brightfield Cell Microscopy Images. Ali et al, SLAS Disc. 2021

Input: brightfield microscopy image **x**

Target: nuclear masks **y**

Representation

- Intensity, texture features, …

Loss function

- Mean squared error
- Pixel accuracy
- Object accuracy
- Count accuracy

Covariance function

- From features

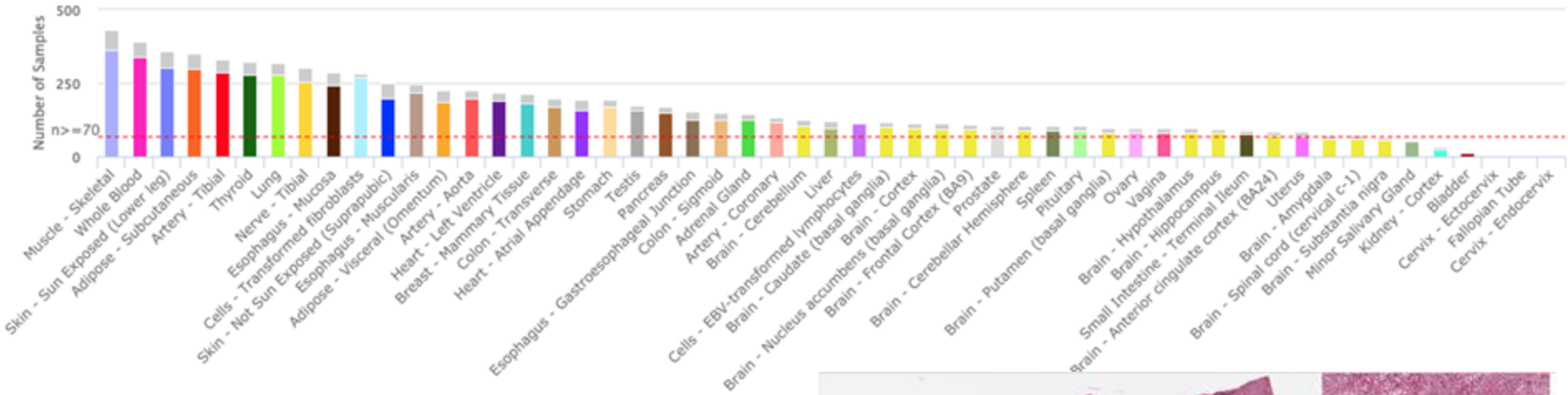
wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for histopathology images

Dataset of human tissues from healthy donors

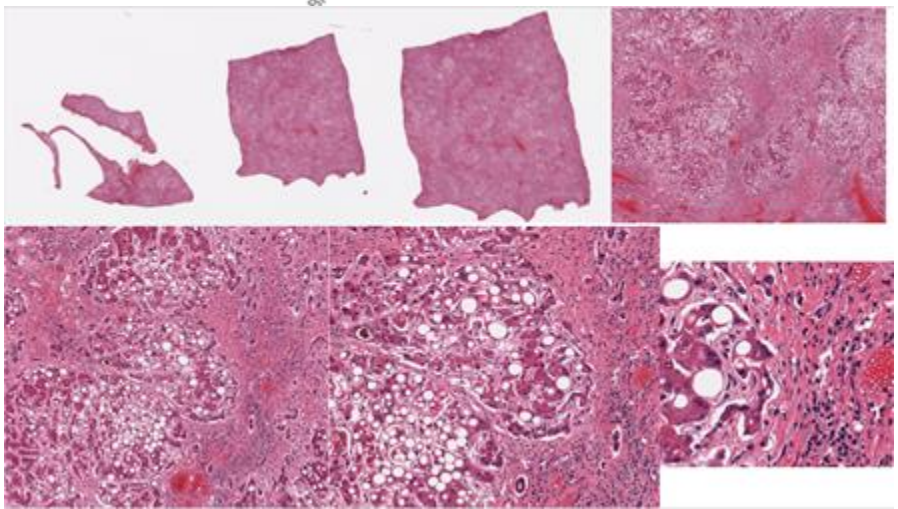
Genotype + bulk RNA expression data.



44 tissues total.

Median of 15 samples per donor

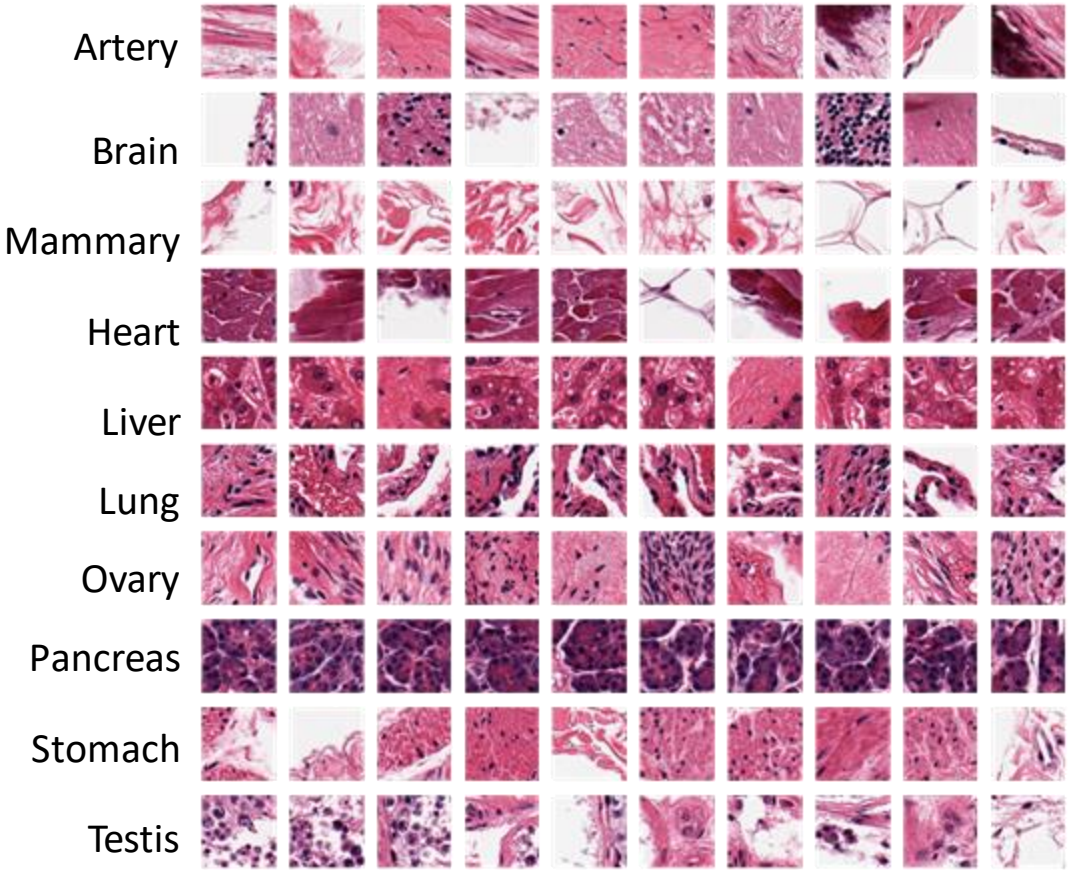
Median of ~150 donors per tissue



wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for histopathology images



10 random image patches

Input: histopathology image **x**

Target: gene expression **y**

Representation?

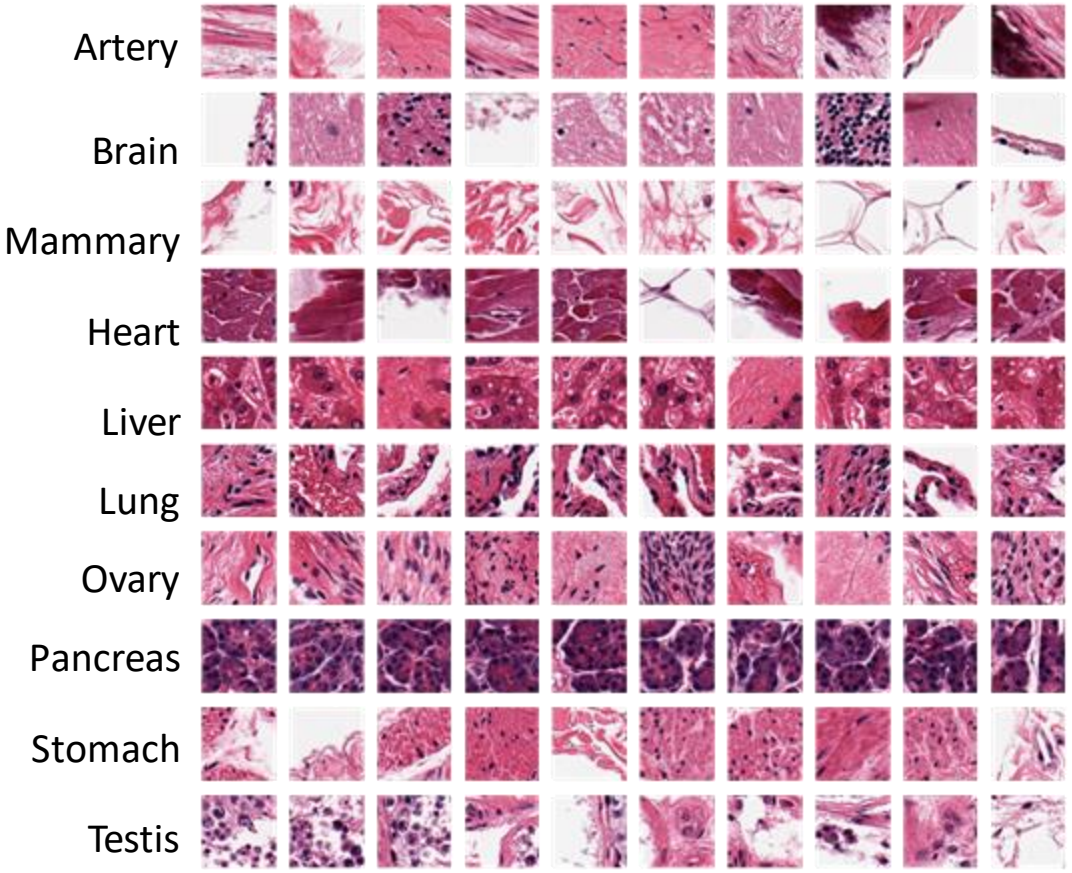
Loss function?

Covariance function?

wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

Deep learning for histopathology images



10 random image patches

Input: histopathology image **x**

Target: gene expression **y**

Representation

- Tissue type
- Features and filters
- Immune cells
- Nuclei count, shape
- Fat deposits
- Blood vessels

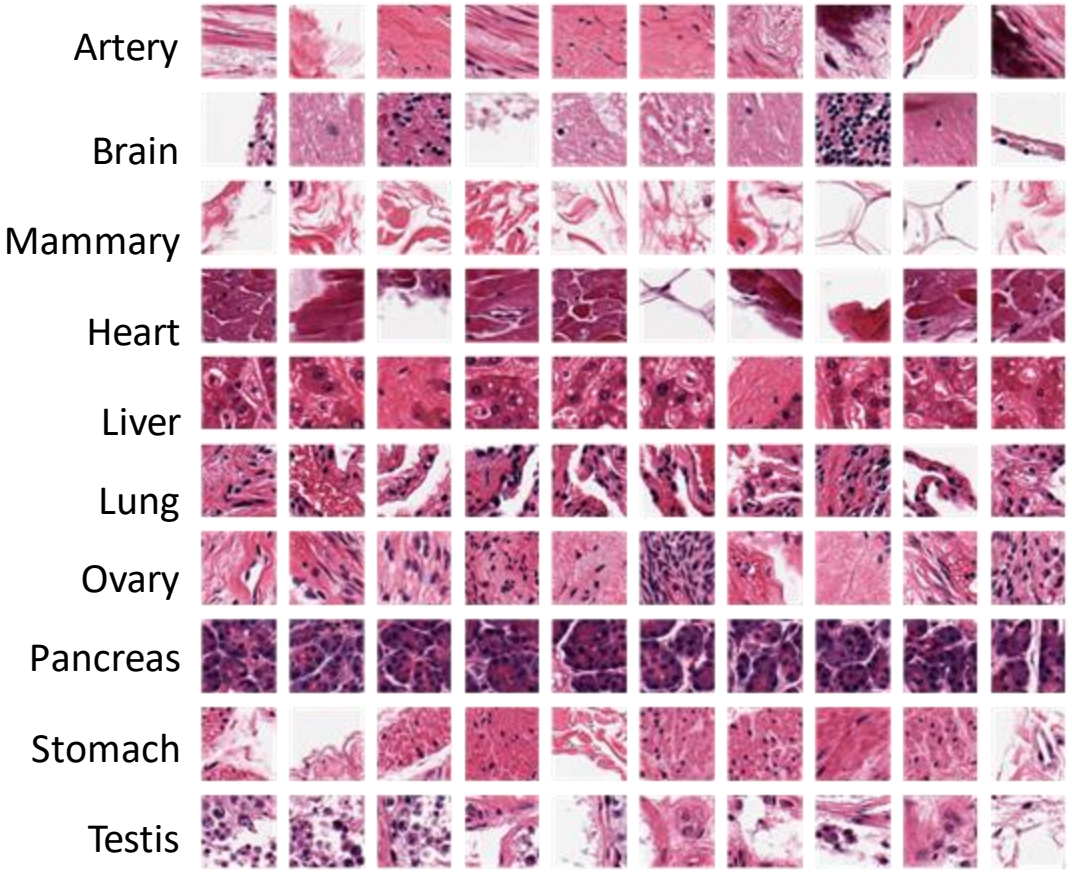
Loss function?

Covariance function?

wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

Deep learning for histopathology images



10 random image patches

Input: histopathology image **x**

Target: gene expression **y**

Representation

- Tissue type
- Features and filters
- Immune cells
- Nuclei count, shape
- Fat deposits
- Blood vessels

Loss function

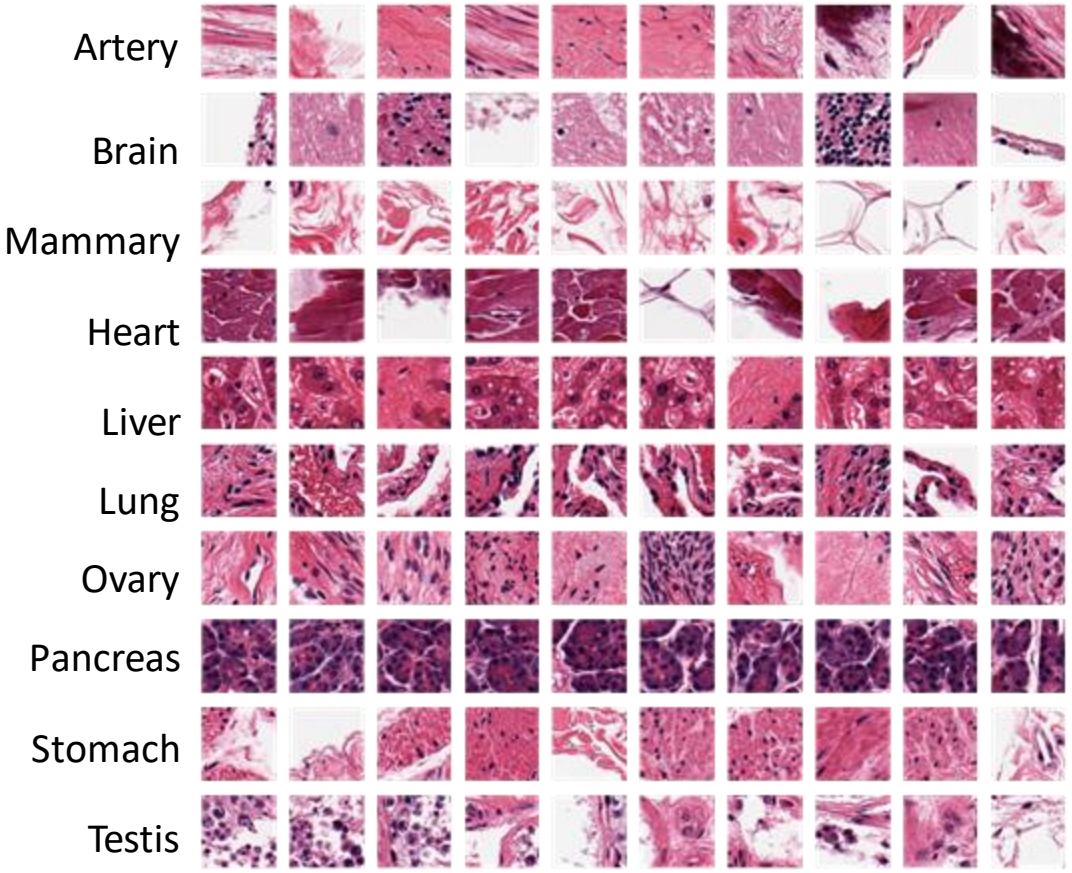
- Mean squared error

Covariance function?

wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

Deep learning for histopathology images



10 random image patches

Input: histopathology image **x**

Target: gene expression **y**

Representation

- Tissue type
- Features and filters
- Immune cells
- Nuclei count, shape
- Fat deposits
- Blood vessels

Loss function

- Mean squared error

Covariance function

- Feature-based

wellcome trust sanger institute

TARTU ÜLIKOOL 1632 arvutiteaduse instituut

Deep learning for histopathology images

Natural approach: find latent representation in image and gene expression space, and see if there is a link

Input: histopathology image **x** 

Target: gene expression **y**

Representation

- Tissue type
- Features and filters
- Immune cells
- Nuclei count, shape
- Fat deposits
- Blood vessels

Loss function

- Mean squared error

Covariance function

- Feature-based

Deep learning for histopathology images

Natural approach: find latent representation in image and gene expression space, and see if there is a link

Naturally lazy (positively) approach: re-use a latent image representation, use PCA for gene expression

Input: histopathology image **x** 

Target: gene expression **y**

Representation

- Tissue type
- Features and filters
- Immune cells
- Nuclei count, shape
- Fat deposits
- Blood vessels

Loss function

- Mean squared error

Covariance function

- Feature-based

Deep learning for histopathology images

Natural approach: find latent representation in image and gene expression space, and see if there is a link

Naturally lazy (positively) approach: re-use a latent image representation, use PCA for gene expression

- Take Inceptionet v3 trained on millions of images
- Add 1024-neuron final layer, predict tissue

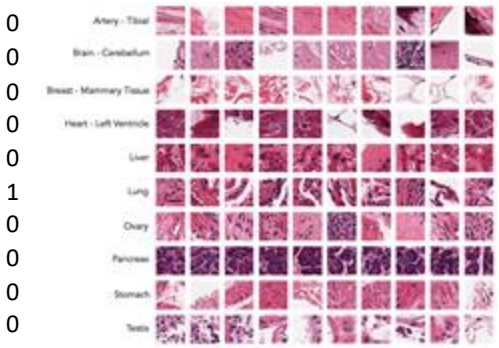
Input: histopathology image **x** 

Target: gene expression **y**

Representation

- Tissue type
- Features and filters
- Immune cells

Input patch 



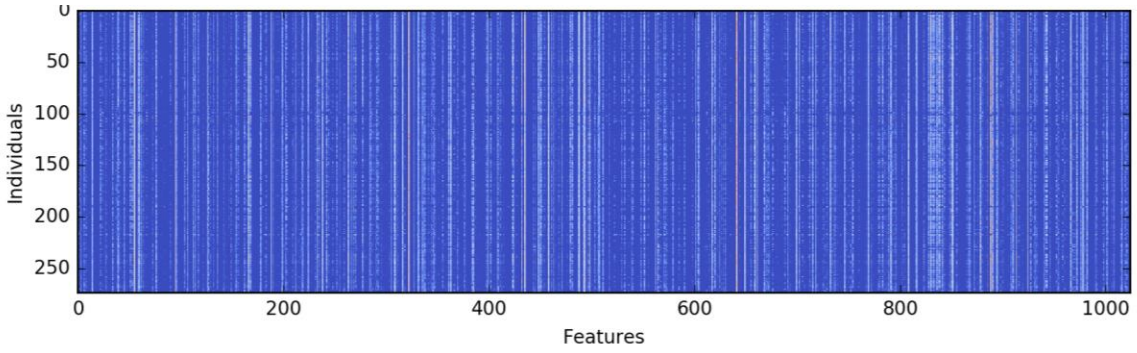
wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for histopathology images

Naturally lazy (positively) approach: re-use a latent image representation, use PCA for gene expression

- Take Inceptionet v3 trained on millions of images
- Add 1024-neuron final layer, predict tissue
- Read final layer activations for each patch, average for each individual
- Run PCA on gene expression matrix, get coordinates for each individual

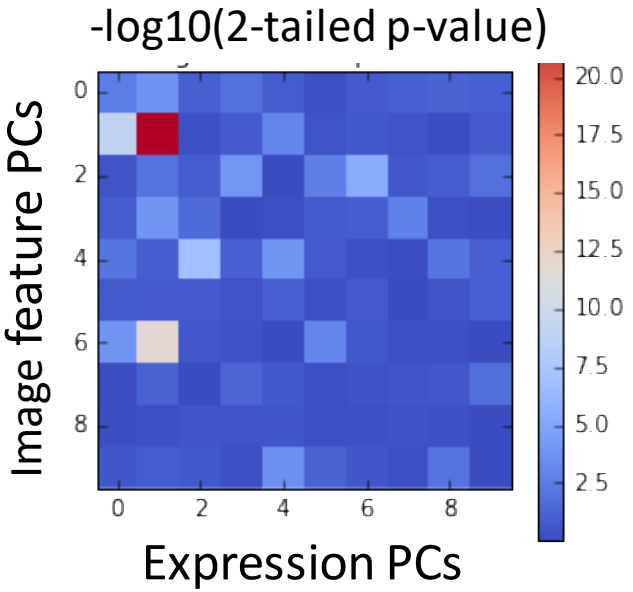


Input: histopathology image **x**

Target: gene expression **y**

Representation

- Tissue type
- Features and filters
- Immune cells

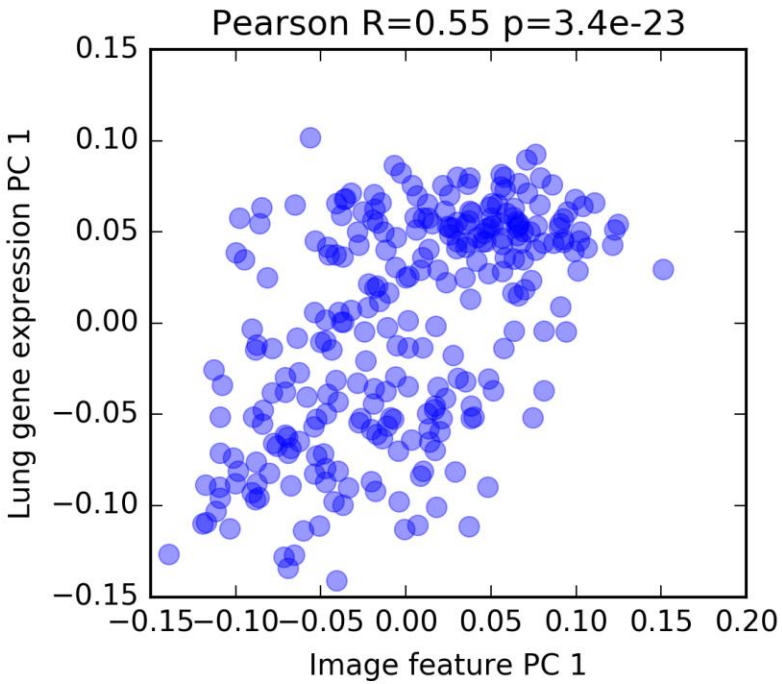


wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for histopathology images

Naturally lazy (positively) approach: re-use a latent image representation, use PCA for gene expression



Input: histopathology image **x**

Target: gene expression **y**

Representation

- Tissue type
- Features and filters
- Immune cells

wellcome trust sanger institute

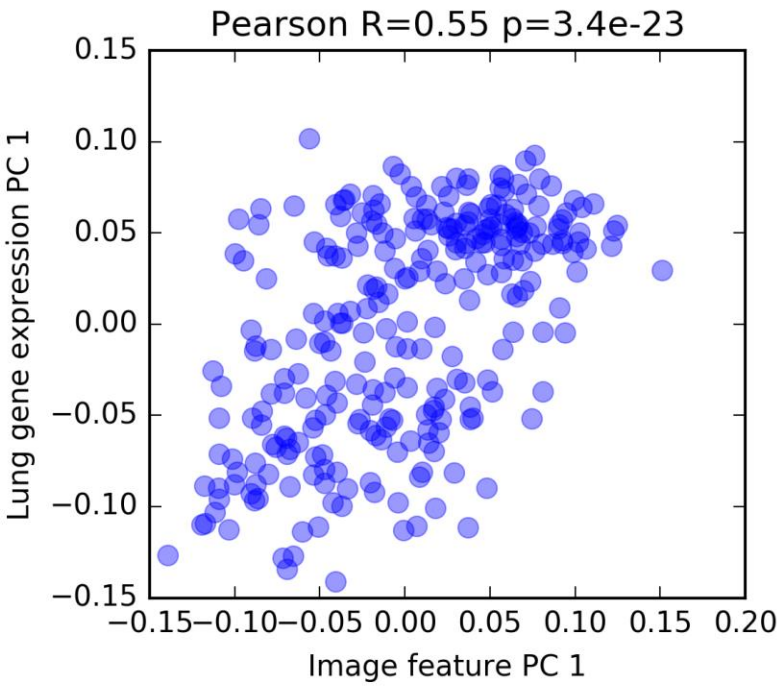
1632 TARTU ÜLIKOOL arvutiteaduse instituut

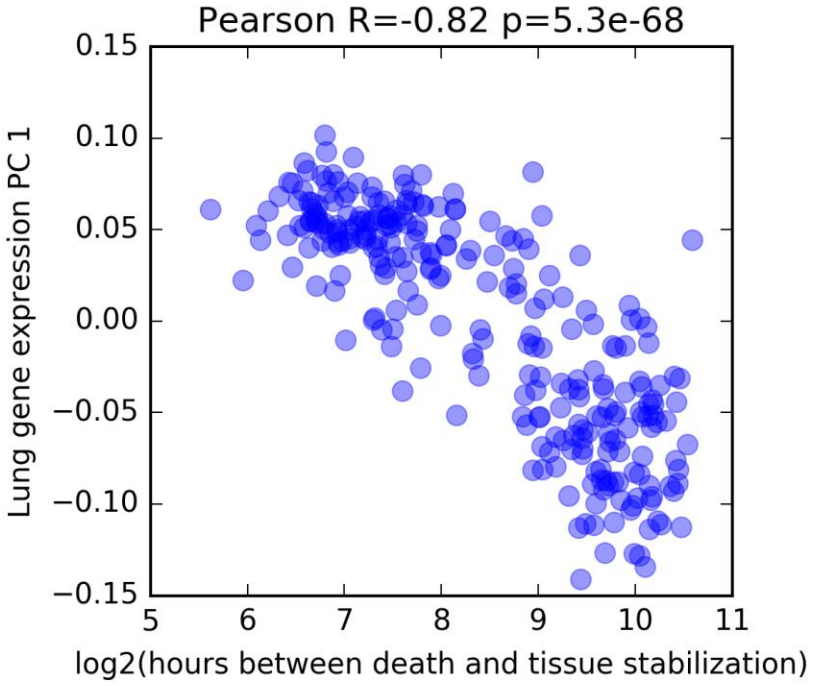
Deep learning for histopathology images

Naturally lazy (positively) approach: re-use a latent image representation, use PCA for gene expression

Input: histopathology image **x** 

Target: gene expression **y**

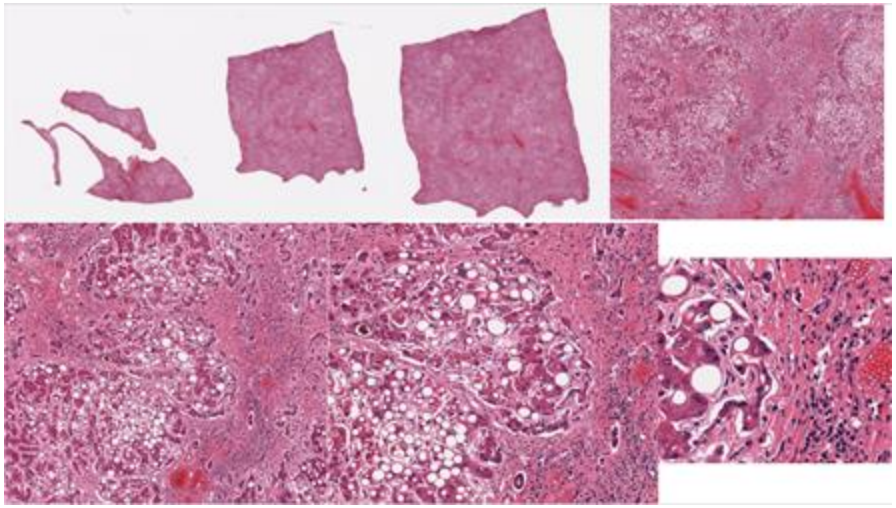




wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for histopathology images



Key insight: huge pre-trained networks can be downloaded and applied directly or finetuned on smaller datasets to perform well on a new task in an unrelated domain

Example papers:

A generalized deep learning framework for whole-slide image segmentation and analysis. Khened et al., Sci Rep 2021.

Input: histopathology image **x** 

Target: gene expression **y**

Representation

- Tissue type
- Features and filters
- Immune cells
- Nuclei count, shape
- Fat deposits
- Blood vessels

Loss function

- Mean squared error

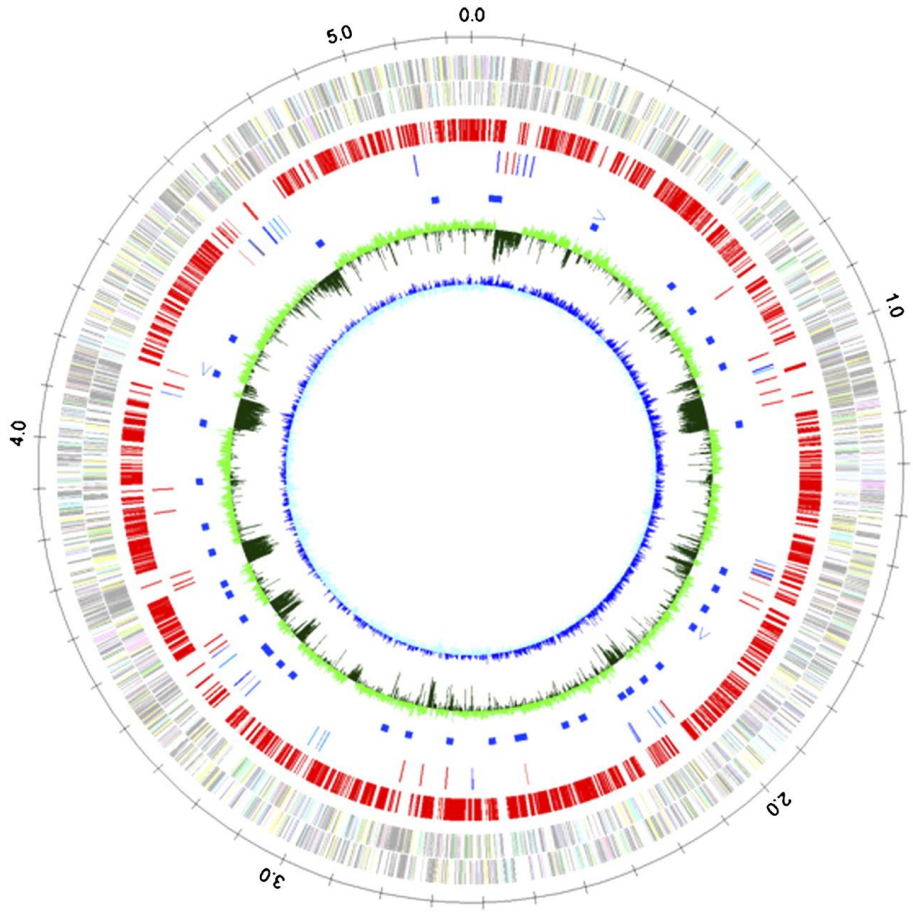
Covariance function

- Feature-based

wellcome trust sanger institute

1632 TARTU ÜLIKOOL arvutiteaduse instituut

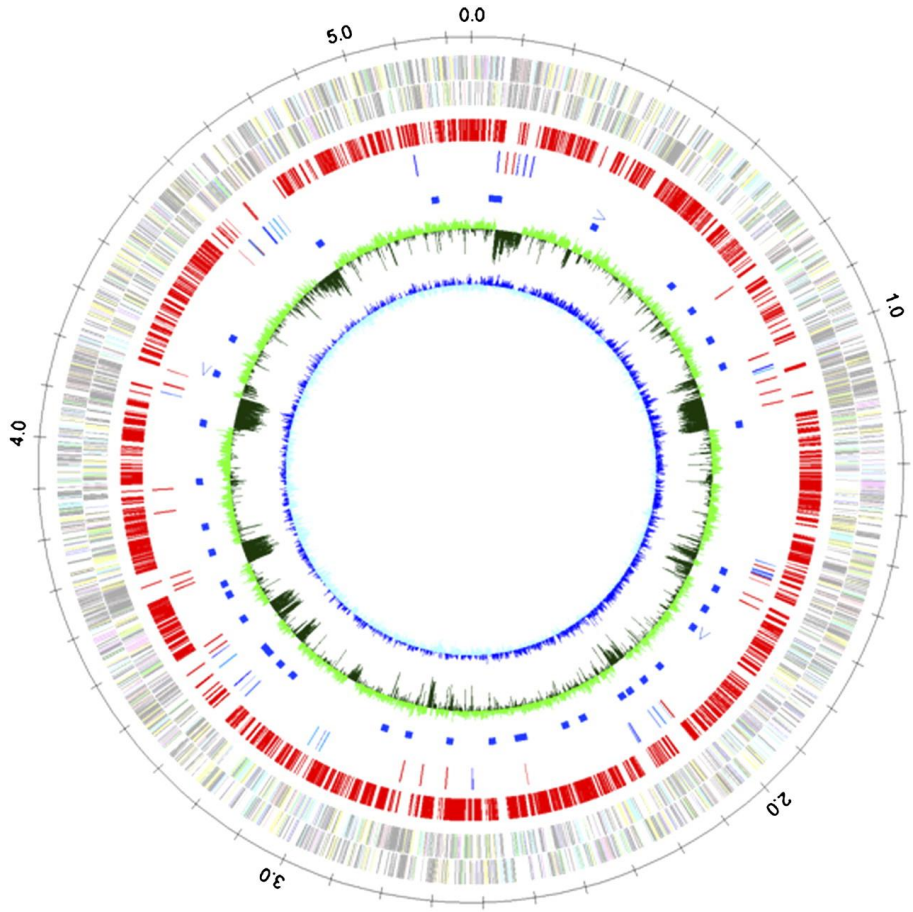
Deep learning for antibiotic resistance



Input: bacterial genome **x**
Target: killing antibiotic concentration **y**
Representation?
Loss function?
Covariance function?

wellcome trust sanger institute
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for antibiotic resistance



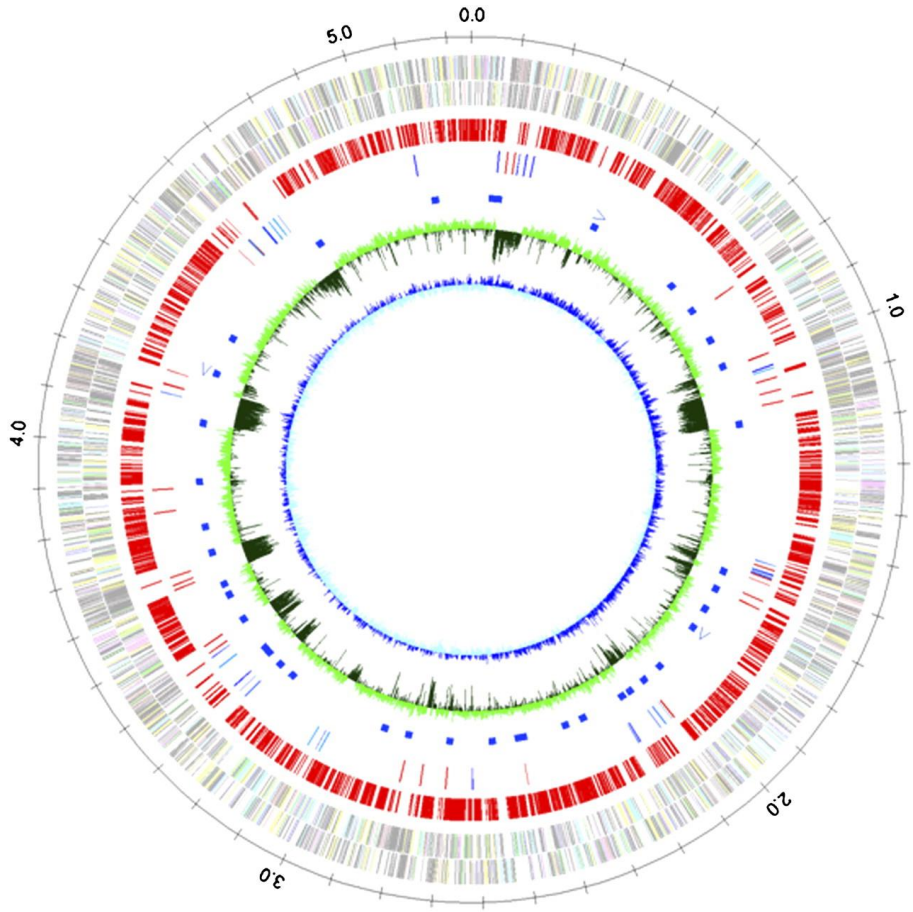
Input: bacterial genome **x**
Target: killing antibiotic concentration **y**
Representation

- Gene content
- Genetic variants
- DNA k-mers

Loss function?
Covariance function?

wellcome trust sanger institute
TARTU ÜLIKOOL 1632 arvutiteaduse instituut

Deep learning for antibiotic resistance



Input: bacterial genome **x**
Target: killing antibiotic concentration **y**
Representation

- Gene content
- Genetic variants
- DNA k-mers

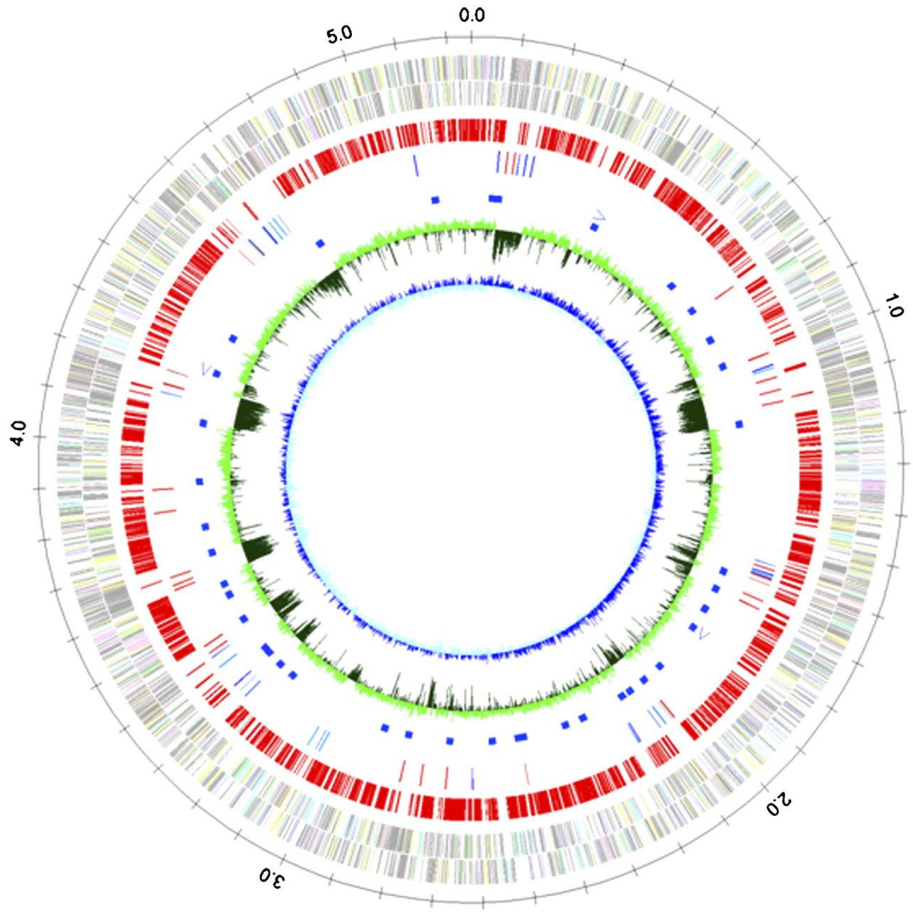
Loss function

- Mean squared error
- Classification accuracy at clinical concentration

Covariance function?

wellcome trust sanger institute
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for antibiotic resistance



Input: bacterial genome **x**
Target: killing antibiotic concentration **y**

Representation

- Gene content
- Genetic variants
- DNA k-mers

Loss function

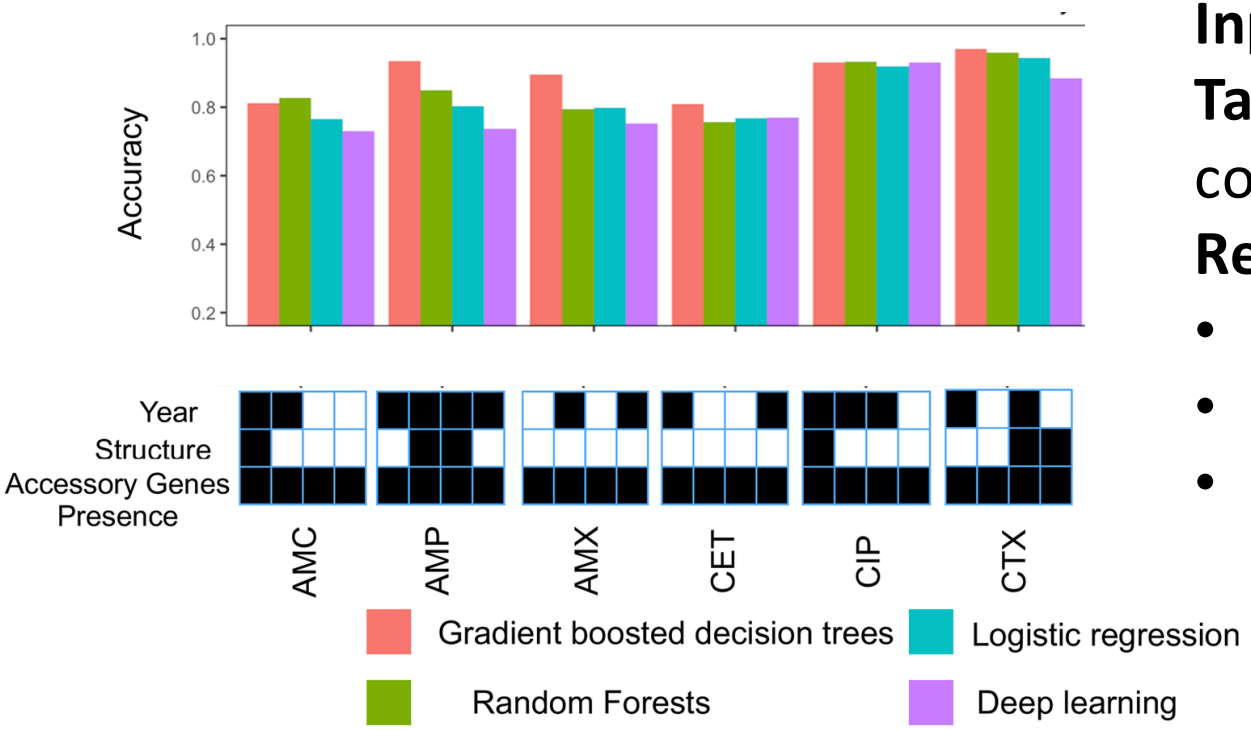
- Mean squared error
- Classification accuracy at clinical concentration

Covariance function

- Gene sharing
- Sequence sharing

wellcome trust sanger institute
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for antibiotic resistance



Input: bacterial genome **x**

Target: killing antibiotic concentration **y**

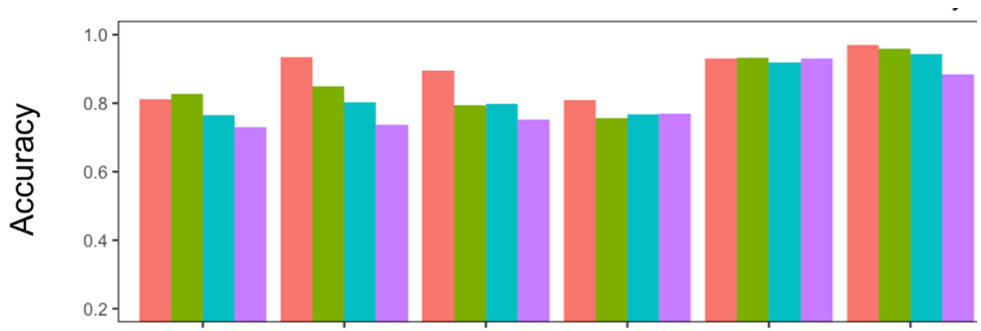
Representation

- Gene content
- Genetic variants
- DNA k-mers

wellcome trust sanger institute

TARTU ÜLIKOOL arvutiteaduse instituut 1632

Deep learning for antibiotic resistance



Key insight: fully connected neural networks can be a quick way to encode classification and regression models. Their performance is not automatically better than simpler and more interpretable approaches.

Example papers:
Machine learning prediction of resistance to subinhibitory antimicrobial concentrations from *Escherichia coli* genomes. Benkwitz-Bedford et al., Msystems 2021

Input: bacterial genome **x**
Target: killing antibiotic concentration **y**
Representation

- Gene content
- Genetic variants
- DNA k-mers

Loss function

- Mean squared error
- Classification accuracy at clinical concentration

Covariance function

- Gene sharing
- Sequence sharing

wellcome trust sanger institute
1632 TARTU ÜLIKOOL arvutiteaduse instituut

Deep learning for …?

Input: instance **x**
Target: property **y**
Representation?
Loss function?
Covariance function?

End of Part 2. Deep learning for …computational biology

- Deep learning is ubiquitous, with the most useful models learning complex representation from very large scale data
- There are many cool ideas on how to capture particular types of signal that people have already tested
- Implicitly or explicitly, the choices of neural network structure and loss function define the type of signal that can be captured, and properties of output that we value
- Sometimes it is more natural to think in terms of the covariance function (what makes two instances similar), sometimes in terms of features (what are the properties that should matter); conceptually, these are two sides of the same coin

Part 3. Guidance [with old person caveat]

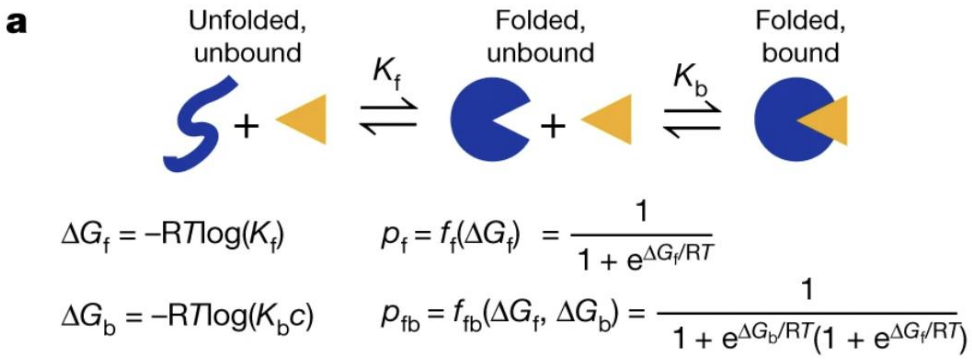
*"1. Anything that is **in the world when you're born** is normal and ordinary and is just a natural part of the way the world works. [Bayes, NHST, GLMs, PCA, decision trees]*
*2. Anything that's **invented between when you're fifteen and thirty-five** is new and exciting and revolutionary and you can probably get a career in it. [SVM, RF, GP, DL]*
*3. Anything **invented after you're thirty-five** is against the natural order of things. [AlphaFold]" - Douglas Adams*

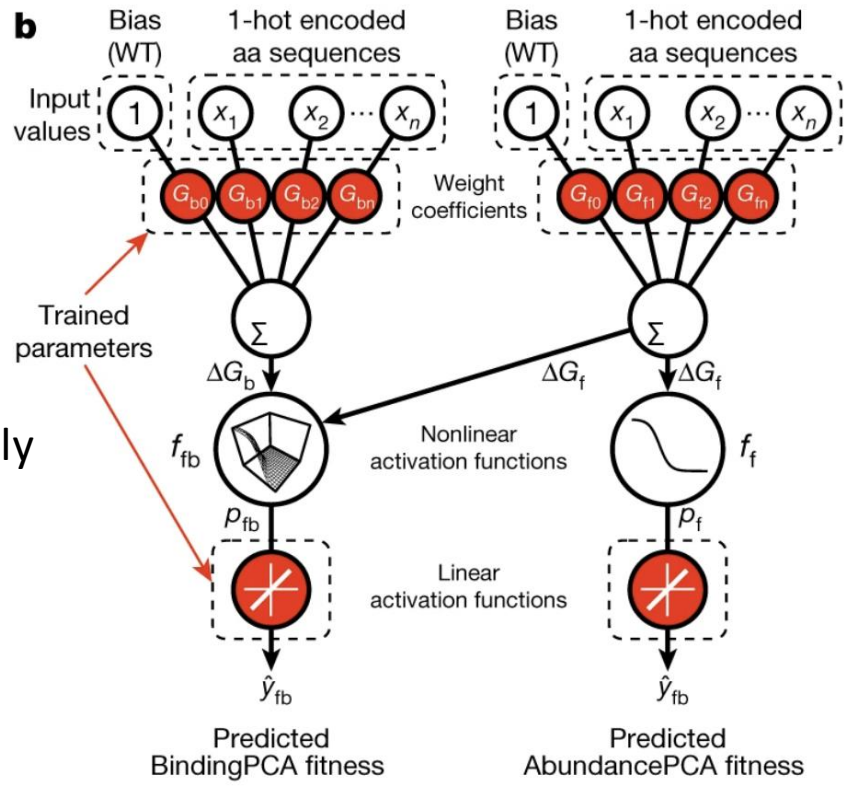
Guidance – when to use deep learning?

- You care about performance above a fixed level, but not understanding [cell counting]
- Large N, large P (lots of data, many parameters) [segmentation]
- Exploration with general purpose computation (see also “automated statistician – grammar over kernels)
- Application of huge pre-trained models (let others do the work; but beware proprietary)

Guidance – when to use deep learning?
General purpose computation and inference

From: Mapping the energetic and allosteric landscapes of protein binding domains





Beatiful, centuries old biochemistry, fit and potentially generalized in an all-purpose compute

wellcome trust sanger institute
TARTU ÜLIKOOL 1632 arvutiteaduse instituut

Guidance – when to use deep learning?

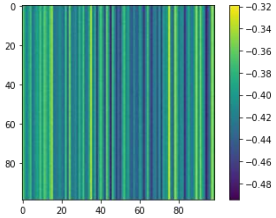
```
#Binding nonlinear layer
binding_nonlinear_layer = State_prob_bound(trainable=False)([binding_additive_trait_layer, folding_additiv
#Binding additive layer
binding_additive_layer = keras.layers.Dense(
  1,
  activation = "linear",
  name = "binding_additive",
  kernel_constraint=Between(0, 1e3)#, bias_constraint=Between(-1, 1)
  )(binding_nonlinear_layer)
### OUTPUT LAYERS
########################################
#Multiplicative layer folding
multiplicative_layer_folding = keras.layers.Multiply()([folding_additive_layer, input_layer_select_folding
#Multiplicative layer binding
multiplicative_layer_binding = keras.layers.Multiply()([binding_additive_layer, input_layer_select_binding
#Sum layer
output_layer = keras.layers.Add()([multiplicative_layer_folding, multiplicative_layer_binding])
#Create keras model defining input and output layers
model = keras.Model(
  inputs = [input_layer_select, input_layer_folding, input_layer_binding],
  outputs = [output_layer])
# Compile model
opt = keras.optimizers.Adam(learning_rate = learn_rate)
#Compile the model
model.compile(
  optimizer = opt,
  loss = 'mean_absolute_error')
```

Guidance – when to use deep learning?

- You care about performance above a fixed level, but not understanding [cell counting]
- Large N, large P (lots of data, many parameters) [segmentation]
- Exploration with general purpose computation (see also “automated statistician – grammar over kernels)
- Application of huge pre-trained models (let Google handle carbon footprint – Inceptionet, Alphafold, Enformer, NucleAIzer, SpliceAI; beware proprietary)

Guidance – where we've seen dragons

- Smallish N, smallish to large P (some data, some to many parameters)
- Playing with hammers without understanding aspects of nails: 1) what you want out of the model 2) if the model is doing well 3) what the outliers are
- Assuming published methods are great and correct [do reimplement, "reasonable starting point"]
- Not making tens of plots rooted in raw data
- Not thinking deeply about the right representation – features, covariance, loss [e.g. evaluating differently]



The End. Suggestions for future ☺

- Enjoy Dima's workshops!
- Develop your skills in deep learning – but as a general purpose modelling toolbox for your needs
- Use this tool judiciously, thinking through the model structure (representation, covariance) and the aim of the work (loss function)
- Convince your duck that you know what you are doing and that it is robust - make lots of plots, use several random seeds, add noise, try other parameter settings