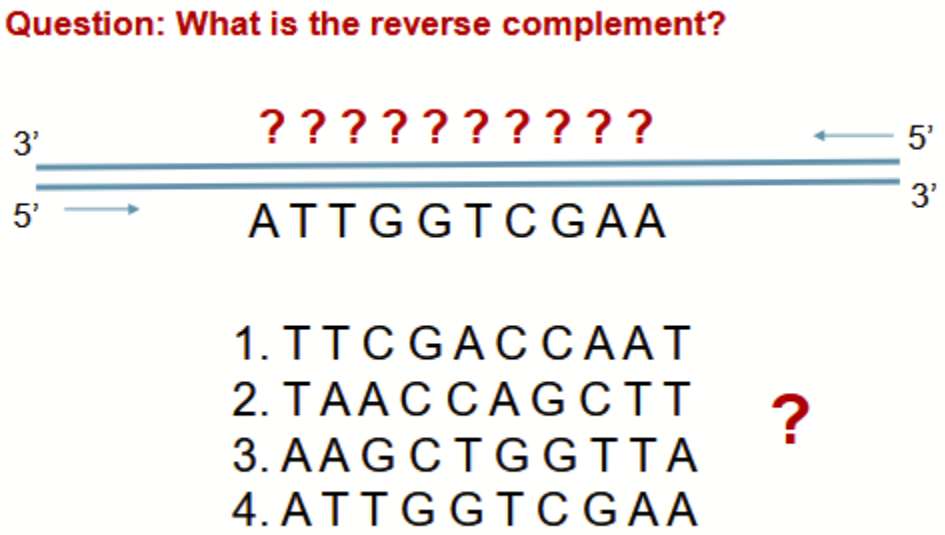
Machine Learning for Genomics Summary FS25

Lecture 1:

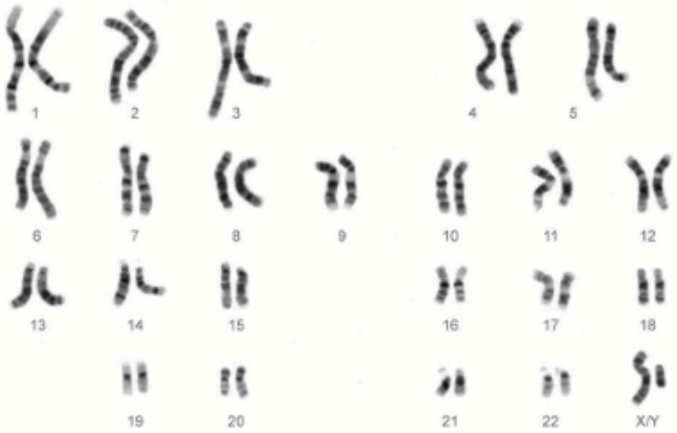
1. **Genome**: Complete set of DNA in an organism, composed of four nucleotides:
    - **Adenine (A)**, **Cytosine (C)**, **Guanine (G)**, **Thymine (T)**.
2. **DNA Structure**:
    - **Double helix** with a sugar-phosphate backbone and nitrogenous bases.
    - **Base pairing**:
        - **A-T** (2 hydrogen bonds), **G-C** (3 hydrogen bonds).
3. **Directionality**:
    - DNA/RNA synthesis occurs in a **5′ to 3′ direction**.
    - Complementary strands are reverse complements (e.g., **AGTC → GACT**).
4. **Reverse Complement**:
   Sequence on one strand matches the reverse of its complement.



Correct Answer is 1, as you still need to write it in 5’3’ way

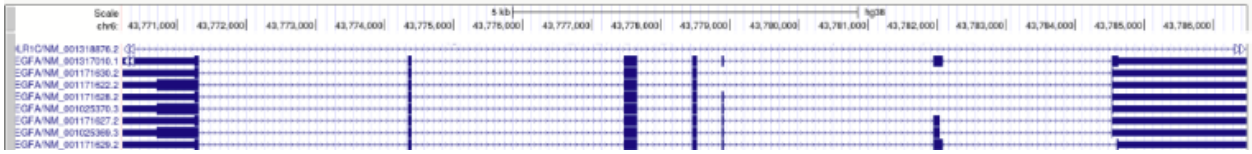
Chromosome 1:

- Length in bp: 248,956,422
- Actual length: 8.3cm
- Space: < two angstroms



Human genome: 2 copies x **3.13 x10^9** base pairs

- CENTRAL DOGMA OF MOLECULAR BIOLOGY: DNA is transcribed into RNA and RNA is translated to proteins.
- Although: Genes are pieces of DNA that get copied into RNA (only coding genes, code for mRNA) ~20K protein coding genes
- Every cell produced roughly only 50% of all possible proteins (due to regulation of transcription)
- Beginning of gene = Transcription start site (TSS)
- End of gene = Transcription End Site (TES)
- Introns & Exons



- Dashed parts are introns which are regions of mRNA, which will be deleted during splicing, not coding for amino acids
- The bold parts (exons) will be present in the so-called mature mRNA after splicing.
- However, exons include bothe protein coding sequences, like the “lone lines in the middle” and the 5’ and 3’ untranslated regions (UTR) signifying the start and end of the gene
- For transcription to be initiated, the polymerase must first gain access to the promoter region at the beginning of a gene
- Promoter regions can be blocked by closed chromatin (because dna is wrapped around histones)!
- Transcription factors can dispose of nucleosomes, opening chromatin and in general regulate transcription
- Transcription factors can bind at the beginning of a gene (promoter regions) or far away from it (enhancers)

- Pioneering transcription factors (e.g., GATA family, NF-Y) can bind condensed chromatin (closely located nucleosomes), open up condensed chromatin or recruit transcription-activating histone modifications (e.g., H3K4me1 by PU.1)

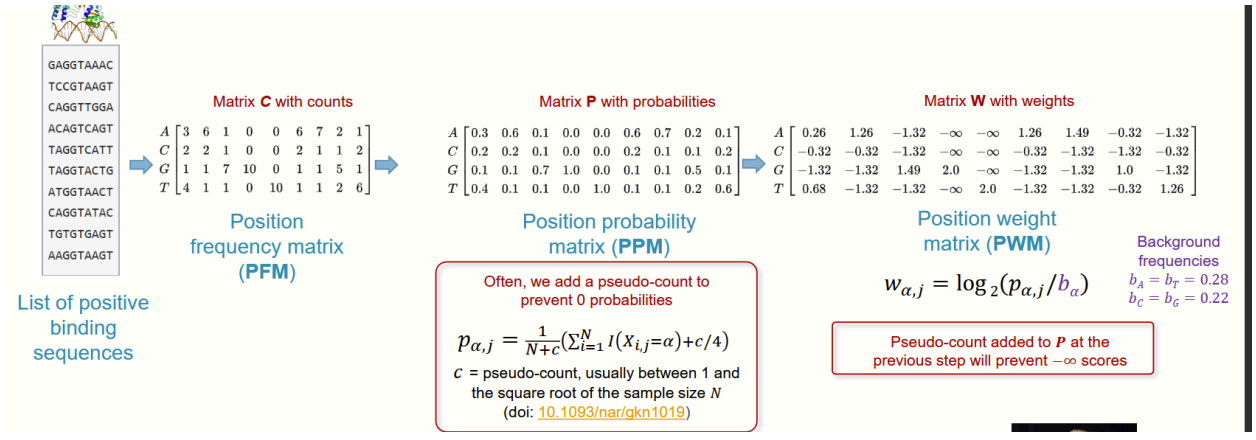
- Transcription factors can affect transcription by:
  - helping dispose nucleosomes from enhancers and promoters
  - helping stabilize RNA polymerase
  - attracting histone acetyltransferases and histone deacetylases (HDACs)
  - attracting other co-activators and co-repressors

- Nucleosomes (histone tails) can be post-transationally modified by acetylation, phosphorylation, methylation etc. and are associated with active gene transcription and gene repression
- Topologically associated domains (TADs) and DNA Looping:
  - Loop Anchors (Cohesin: A **ring-shaped protein complex** responsible for **chromatin looping**)
  - CTCF binds to cohesin to create TADs
  - Promoters and Enhancers within the same TAD are much more likely to interact!
  - Binding of transcription factors (including CTCF and YY1) to DNA influences DNA looping, accessibility of DNA, and modifications of histone tails
- DNA methylation can change affinity of transcription factor binding

LECTURE 2:

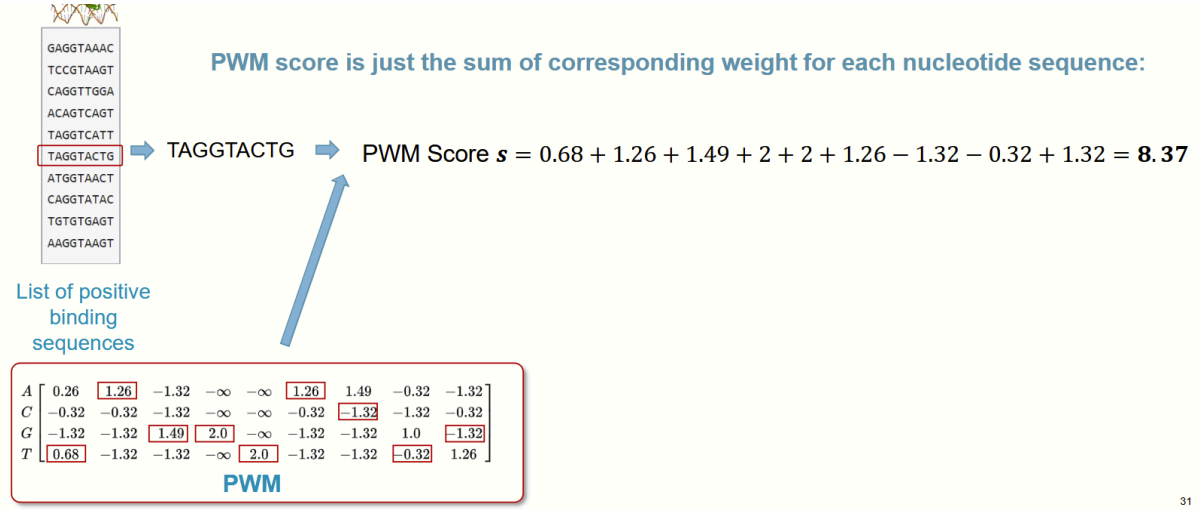
- Chromatin immunoprecipitation and sequencing (ChIP-seq) can be used to experimentally find binding locations of TFs.
- Single nucleotide polymorphism (SNP)
    - SNPs are DNA variants frequent in the population (>1%)
    - Most SNPs don't affect health
- A rare variant (aka mutation)
    - DNA variants which are extremely rare in the population (<<1%)
- Germline variants: mutation in egg or sperm, affects all cells in offspring (heritable (SNPs))
- Somatic variants (occur only in e.g., stomach tissue, non-heritable)
- Coding variant: present in coding exons, could affect protein folding (<5% of variants)
- Noncoding mutations (>95% variants) occur within and outside of genes, but not in coding regions (may influence gene expression, promoter usage and mRNA transcript stability)
- Experiments still needed to understand effect of noncoding mutations
- GWAS (Genome-wide association study):
    - Tests linkage between SNPS and Phenotype
- eQTLs: Expression quantitative trait loci
    - Tests linkage b/w SNPs and gene expression

- Prediction of TF binding motifs (TFBS)
    - Option 1: Model TFBSs of our TF as a list L (Enumeration) (Binding, sequence in L, non-binding not in L)
    - Issue: Some binding sites can be missed, binding affinity not binary
    - Option 2: Model TFBSs of our TF as a summary list L: IUPAC* consensus (binding: sequence in L, no binding sequence not in L)
    - Issue: Binding affinity not binary, information about top nucleotide at each position is not taken into account
    - Option 3: Position weight matrix: PWM (aka position-specific scoring matrix (PSSM))



- Begin by creating PFM, which essentially counts how often each base appears at each position

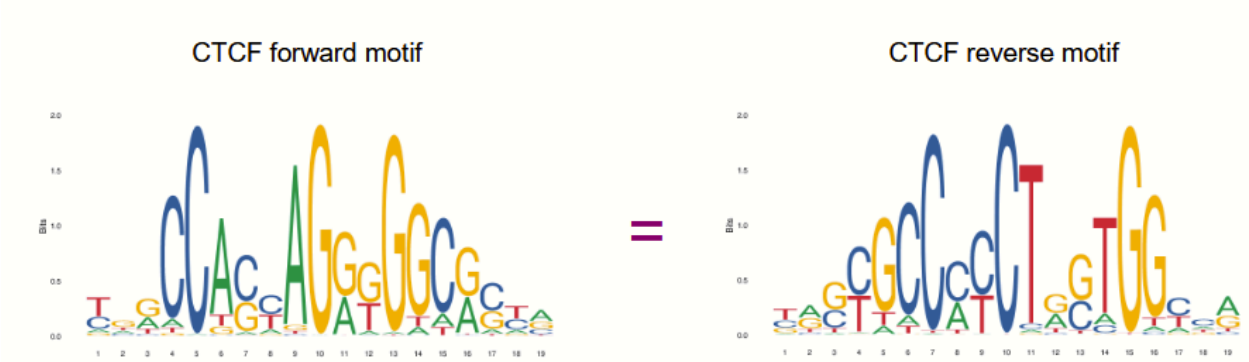
- Normalize to probabilities within columns (normalizing with pseudocount) to prevent 0, so log step in PWM doesn't output negative infinity
- Lastly create weights with log_2 normalization wrt background frequency of bases to create PWM
- Then a PWM score modeling binding affinity:



  - Sequence logos can be created, the total height of which is the information content in bits of the corresponding position

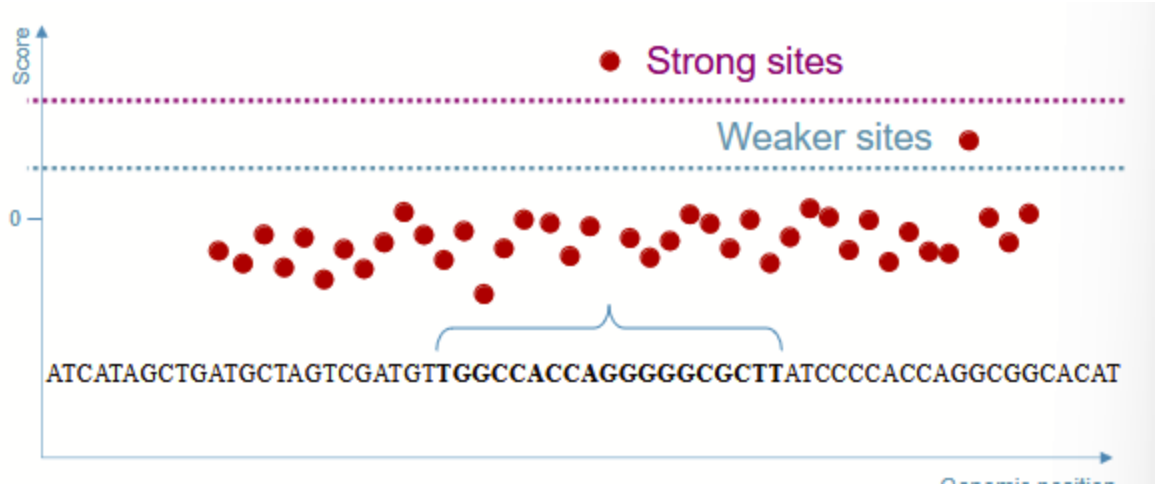
$$IC_j = log_2(4) - H_j = 2 + \sum_{\alpha} p_{\alpha,j} \log_2(p_{\alpha,j}$$

Sequence Logo:



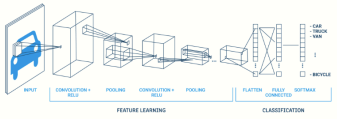
There is no difference between forward and reverse complement of a motif in term of applications

- Finally, Apply a threshold corresponding to your p-value



Option 4: Deep CNN, where each sequence of nucleotides is given a score modeling binding affinity using a regression neural network

- Kernels slide along input features and provide translation equivariant responses known as feature maps
    - Why use multiple filters?
    - In molecular biology, **conformation** refers to the **three-dimensional shape or spatial arrangement** of a molecule, including how its atoms are oriented and organized.
    - Transcription factors can bind to DNA in multiple **conformations**, influenced by factors like chromatin structure, co-factors, and post-translational modifications. Each conformation may have a slightly different DNA-binding preference, resulting in variations in the binding motif.
    - Multiple filters allow the CNN to detect these variations, ensuring the model captures the **full range of binding site patterns** associated with a single TF
    - Use a 1-d convolution with 4 channels (4*26 kernel)
- All neurons in a single depth slice share the same parameters
- Usually Conv-layers are followed by ReLU
- Convolution may include dilation (checkerboard pattern)
- Borders are often handled with padding at the border
- Down-sampling then often occurs with pooling layers (max,avg)



- Deep CNNs are finally regularized with methods like dropout, early stopping
- Convolved signals inherit the topology of original input, so for DNA data, we use 1D CNNs

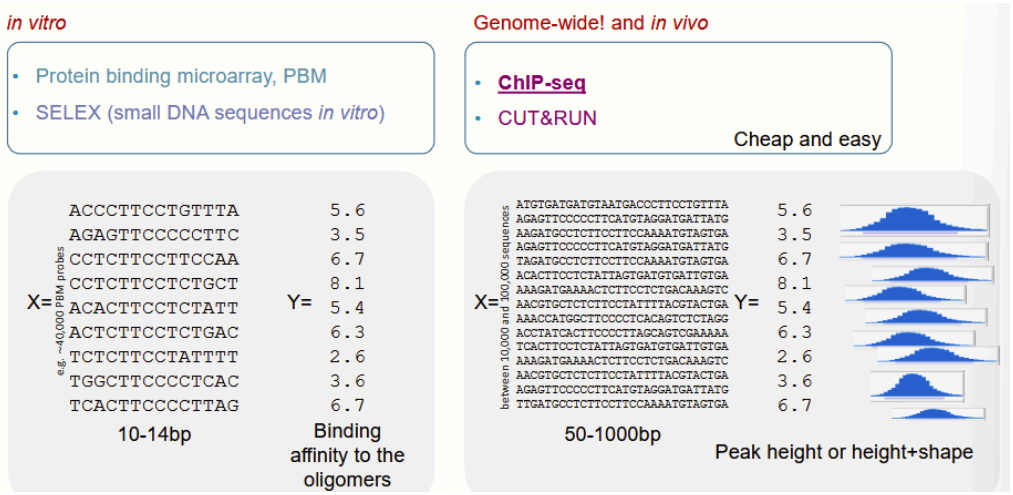
CNN application to DNA:

- Treat 1-hot-encoded DNA sequence as a 1D image and apply CNN to that sequence
- Learned filters will be ~PWM weights

- By implementation of many consecutive layers, CNNs can capture different architectures of motif interactions

How to get data on DNA sequences bound by TF to train models?

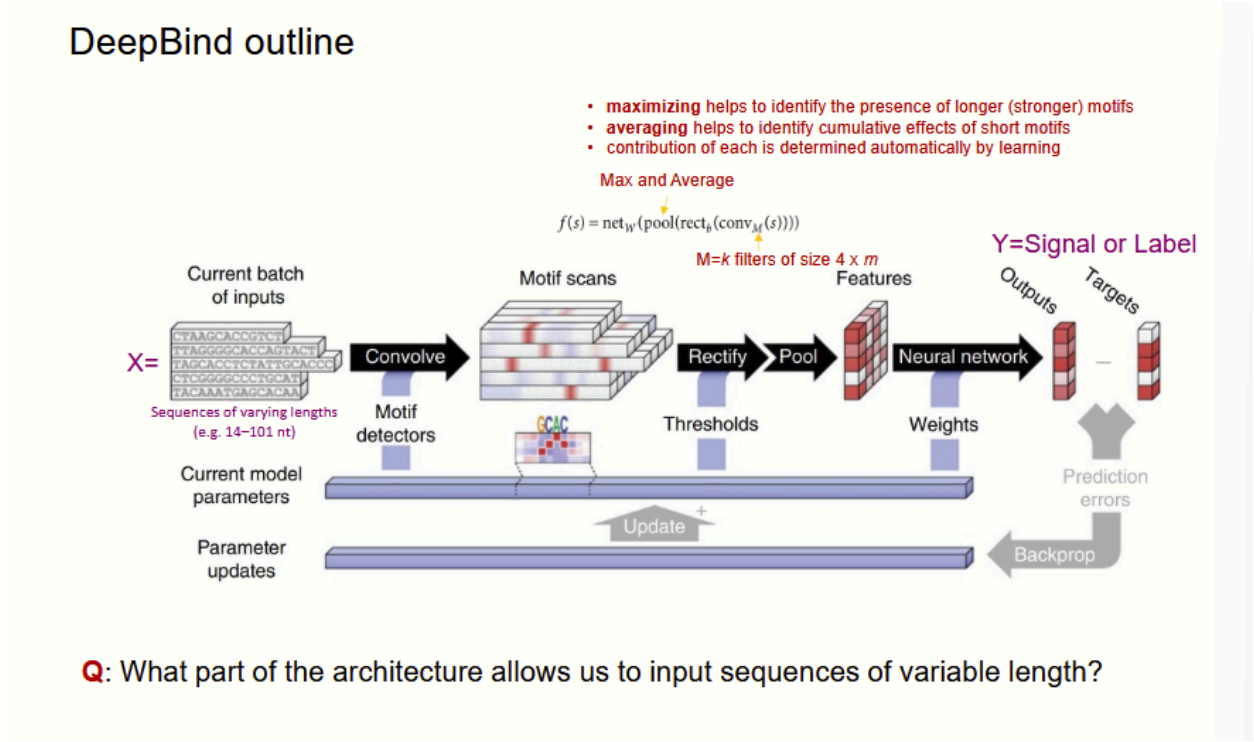
- Protein binding microarray (PBM) (small DNA sequences in vitro)
    - Begin with DNA printed on a microarray
    - Then detect binding events of TF which were tagged before
- SELEX
    - Begin with random DNA fragment pool
    - Incubate with TFs
    - Amplify the DNA fragments bound by TF
    - Use amplified DNA as input to next SELEX round
    - Finally sequence batch
- ChIP-seq
    - Identifies the locations in the genome bound by proteins
    - Not only TF, but also other DNA binding proteins like histones
        - Basically, take chromatin data
        - Fragment into 200 bp parts
        - Isolate all parts with protein of interest using an antibody
        - Isolate chromatin, amplify and sequence the reads
        - The sequenced reads can be mapped to positions on the genome to show which regions of the DNA are highly bound by protein X
    - We assume the peak top of the ChIP-seq density profiles to be the approximate binding position of a TF
    - Peak height describes the approximate frequency of binding



CNNs and ChIP-seq data:

- Using several filters allows CNNs to capture several binding motifs for the main TF and TFs that facilitate its binding
- By using a large receptive field, enabled by parameter sharing, CNNs can capture long-range dependencies, i.e. distant nucleotides important for TF binding.
- Dilated convolutions widen the receptive field even more allowing for more long-range dependencies.

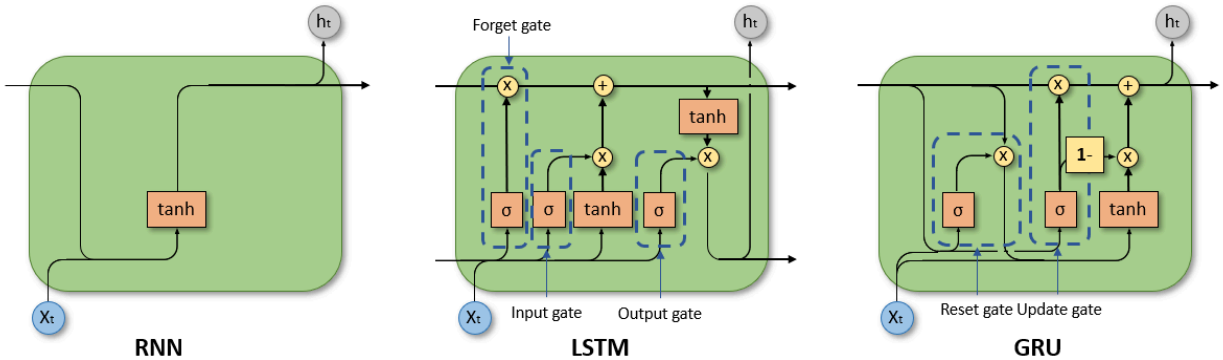
- DeepBind



- Where the pooling layer allows to deal with variable length

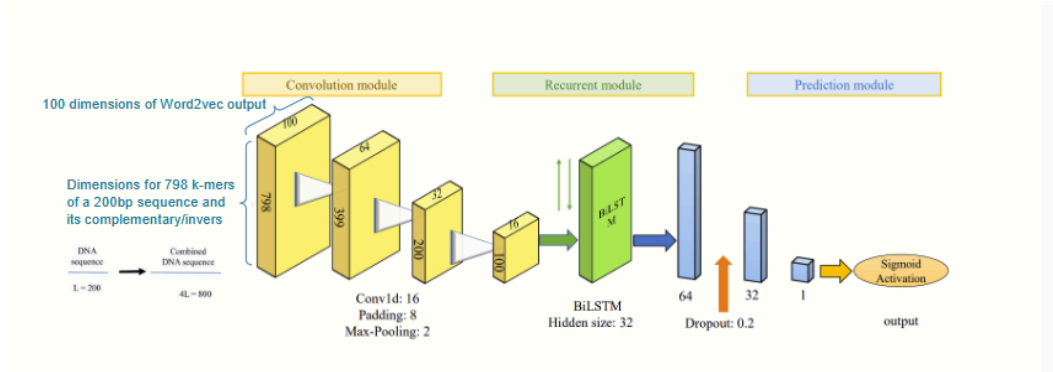
LECTURE 3 - Working with DNA 2

- Transcription factors (TFs) can bring DNA histone acetyltransferases, which acetylate histones, and initiate transcription
- How to improve TF binding motif prediction?
- New Encoding of DNA sequence:
- K-mer encoding of DNA sequence:
    - Instead of using a one-hot encoding as an input, a 2-mer is used where ATTAC is transformed to {AT,TT,TA,AC} (k=2). Length of sequence. n=((l-k)/s+1) s=stride
    - K-mers can be transformed into vector space with word2vec using CBOW
    - Then each k-mer is modeled with a 100-dim vector
    - Word2Vec transformed DNA sequence can be used as an input to CNN
- RNN
    - particularly useful for processing sequential data such bases in a DNA molecule
    - In DNA context, allow Bi-directional hidden state evolution, as it mustn't be causal
    - Positive: Learns sequential information
    - Negative: RNNs are known for exploding and vanishing gradients
- Solution: LSTMs: gated units of LSTM and GRU networks
    - LSTMs also allow gradients to flow unchanged

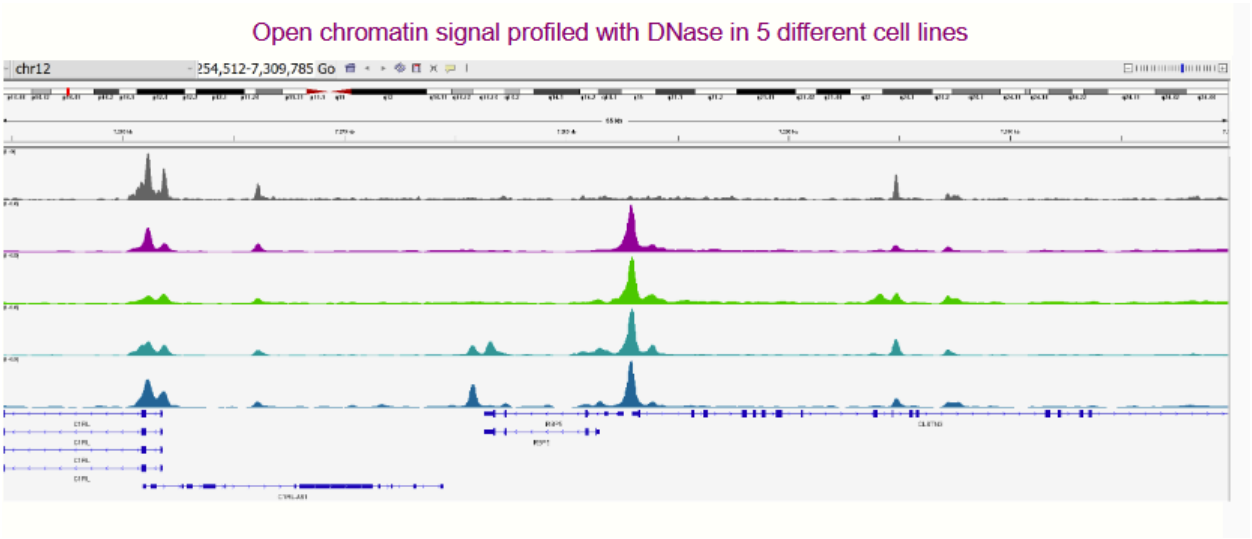


- Bi-directional LSTMs are widely used in DNA analysis
    - Drawbacks of LSTMs and GRU:
    - Difficult to interpret/to understand what LSTMs and GRUs actually learn
    - Long range dependencies remain difficult to capture
    - Difficult to train, although GRUs train better
    - Siamese architecture that accounts for reverse complements

- LSTMs layers can be used after the pooling layer of CNNs to model that motifs follow regulatory grammar governed by physics which dictate the in vivo spatial arrangements and frequencies of combinations of motifs.
- CNN captures regulatory motifs, LSTM captures regulatory grammar



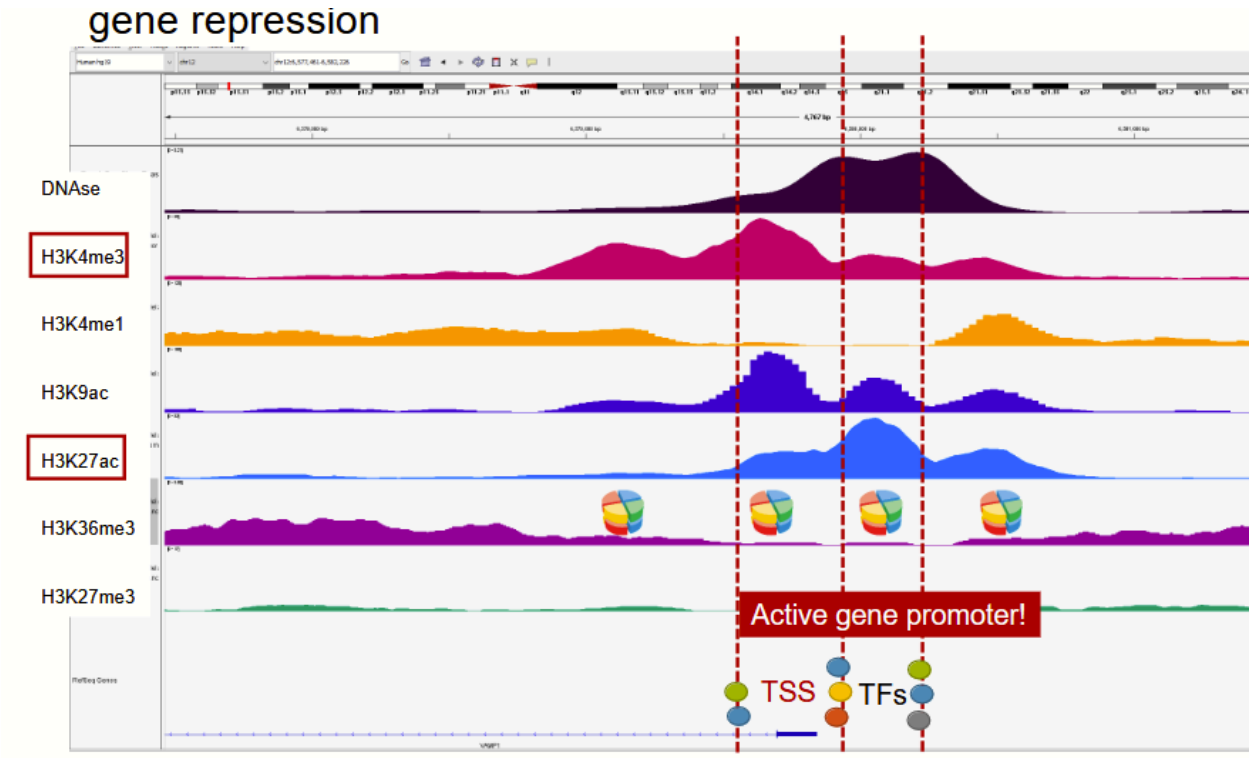
- Attention! Is the ability to actively process specific information in the environment while tuning out other details
    - Self-attention is an attention mechanism relating different positions of an input sequence in order to compute a representation of the sequence
    - **RNNs have trouble memorizing the past and knowing what to forget about the past. Attention allows to view the input as a whole, using all information available without forgetting**
    - By adopting the attention mechanism, TBiNet could focus more on DNA sequence region containing TF binding motifs and thus improve its performance
- Prediction of TF binding using information about open chromatin!
- Experimental methods to profile open chromatin
    - DNase-seq (DNase I hypersensitive sites sequencing): Using DNase I (a nuclease which selectively digests nucleosome-depleted DNA), regions that are hypersensitive to DNase I allow us to find nucleosome-depleted regions aka open chromatin
    - ATAC-seq (Assay for Transposase-Accessible Chromatin using sequencing): Using a hyperactive mutant Tn5 Transposase to insert transposable markers with specific adapters into open chromatin, PCR can be used to amplify sequences adjacent to transposons, allowing to determining open chromatin
    - Alternatives: FAIRE-seq and MNase-seq



- Prediction of TF binding using open chromatin data:
    - CENTIPEDE: Bayesian Mixture Model based on PWM, sequence conservation, DNAse-seq (Open chromatin) and optionally histone marks.
        - Adding DNase accessibility improves model performance
    - Arvey's method: SVM classifier using k-mers + DNAse-seq
        - Adding DNase info improved prediction performance
    - FactorNet: hybrid CNN + RNN using DNA sequencing data + DNAse-seq + others
        - No ablation study
    - Attention-based: DeepGRN: one-hot encoding of DNA sequence + DNAse-seq + others on CNN+RNN arch, ensembling with attention modules.
        - 1st attention module measures important of different regions along the sequential input
        - 2nd attention modules measures pairwise attention to attend the importance between each pair of positions
        - Attention helps better predict signals in low DNAse-seq regions!!!
- 35bp mappability uniqueness: Mappability refers to the ability of sequencing reads of a given length (e.g., 35 bp) to be **uniquely aligned** to a single location in the reference genome

LECTURE 4: Working with Chromatin

- Histone marks: Modifications of histones that indicate cell machinery which genes to transcribe and/or to start transcription
- Histone modifications are associated with active gene transcription and gene repression



- As can be seen in the picture, histone mods are e.g. associated with the start of transcription / areas of open chromatin
- Active gene promoters:
    - Associated with open and accessible chromatin, where DNase-seq peaks are flanked by regions of histone modifications peaks

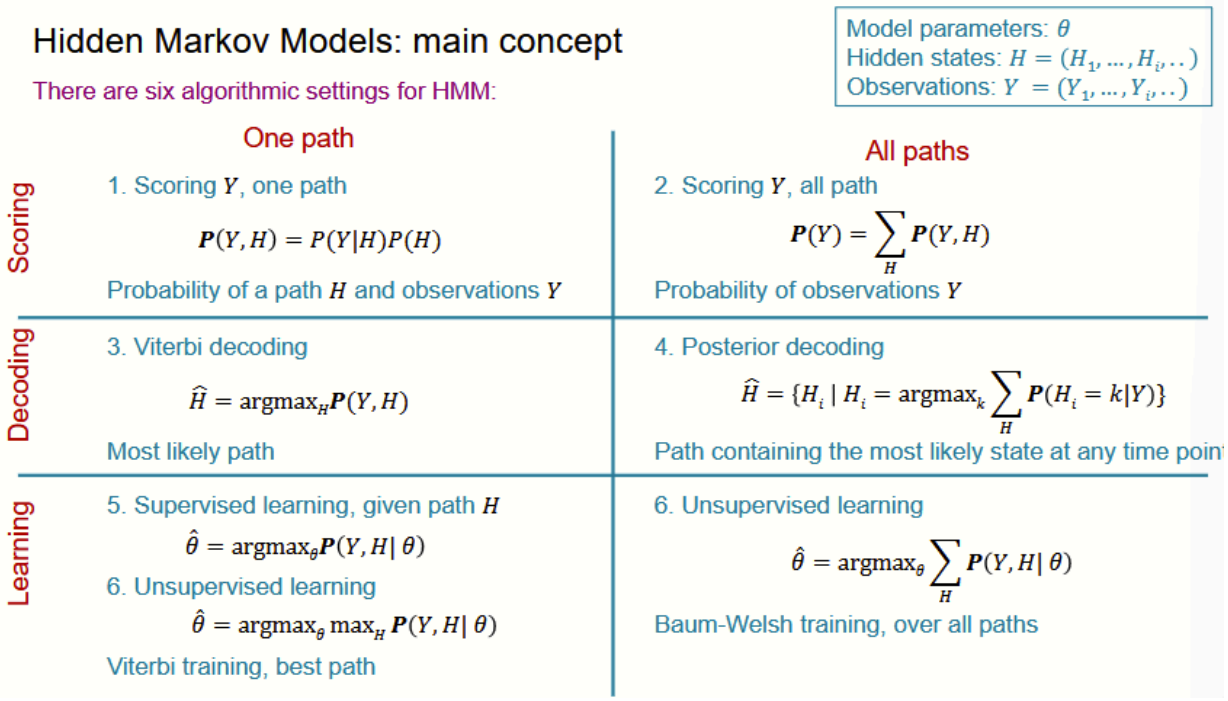
    - H3K27ac (active enhancer) H3K4me3 (active promoter)
- Repressed gene:
    - Associated by closed and inaccessible chromatin
    - Marked b H3K27me3 or M3K9me3
- TF binding information alone is not enough for good prediction of gene expression, chromatin states and histone modifications are paramount

RQ: Predicting cell-type specific effects of variants (mutations) on chromatin accessibility

- Basset (2016): 3 Conv-layers, ReLU and max pooling
    - 300 filters for 1st layer, 200 for rest
    - Stoch-GD, batch normalization
    - FC layers
    - Prediction: Chromatin accessibility (open/closed)
    - Why use multitasking i.e. predict 160 cell lines using 1 model? By multitasking, we may learn common “grammar” better which could help predict variants’ chromatin accessibility better
- DeepSEA (2015) CNN arch. Using DNAse-seq and ChIP-seq histone data for training
    - DeepSEA predicts TF Binding, DNase I sensitivity and histone mark profiles
    - Good performance
    - Predictions which are not cell-type specific still missing.
    - Information about gene expression is not used during training
    - CNN-based ExPecto and Basenji use gene expression during training

Statistical ML models to annotate the genome:

- Assume that histone modifications are associated with certain chromatin ‘states’ aka epigenetic code
- Can we train a model to predict the states of the chromatin in an unsupervised way?
- Annotating Chromatin with “hidden states”: HMM
- Markov property: P(H_{n+1}|H_1,H_2,...) = P(H_{n+1}|H_n)
    a. Moreover, P(H_1,H_2,....) can be factored P(H_1)P(H_2|H_1)P(H_3|H_2)....
    b. Emission probabilities: probability of observing outcome y_n given hidden state H_n :P(Y_n|H_n)
    c. P(Y_n|H_1,H_2,...) = P(Y_n|H_n)
    d. Transition probability: P(H_{n+1}=X_j|H_n=X_i)
- In HMM we try to learn H_{1:n} by observing events Y_{1:n}



1. Scoring: Find the probability of H given Y and sum over all observations Y (by factoring into P(Y|H)*P(H)
2. Decoding: Find most likely states \hat{H} based on argmax_H P(Y,H)
   Biological Intuition:H= functional units of genetic code (intron, exon,active promoter,...) and the Y= Histone modifications
   Watch out: Posterior decoding may give invalid sequence of states
3. Learning: Learn model parameters (=emission & transition probs) given the “optimal” path H found in step 2
   a. Initalise
   b. Iterate (1-2) until P(Y|\theta) converges

- ChromHMM: automates chromatin-state discovery and characterization
- To select the number of states (1) use your biological intuition and (2) score with log-likelihood minus a BIC penalization

Latent Dirichlet Allocation (LDA): dirichlet as distributions are modeled with dirichlet distribution

- Topic modeling: an alternative to HMM with “soft” clustering instead
- Each observation is taken to be a document which is modeled as a convex mixture of k topics; topics determine which words (distributions) are used in each document
- LDA is an important topic modeling technique
- In genomics, the SNP positions are documents
- Topics are the hidden functional states {enhancer, open chromatin, not functional,...}
- LDA will output a mixture (soft) assignment to the topics (states) FUN-LDA
- FUN-LDA often overperforms ChromHMM (but not plain DNase (open chromatin) signal)

LDA and HMM do not use DNA sequence (instead only chromatin marks)

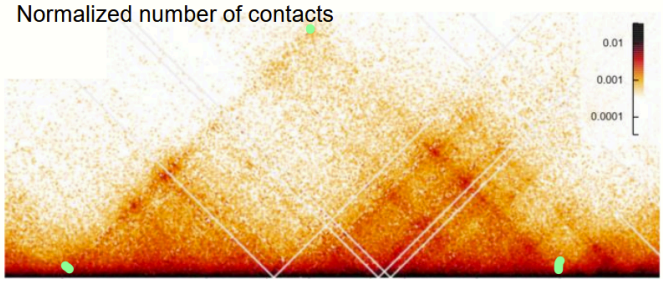
LECTURE 5 - Chromatin and protein folding
Prediction of 3D structure of Chromatin

- For TADs to occur, the CTCF factors must be convergently oriented (see arrow)

- I.e. Non-coding mutations affecting the 3D structure of chromatin are likely to change gene expression
- Remember, Promoters and enhancers located within the same TAD are much more likely to interact
- Predicting the 3D structure of chromatin can help us know the topology which (1) can tell us which genes are affected by mutations in non-coding DNA and (2) which genes are affected by CTCF sites being disabled
- In general, if a TFBS and CTCF can result in different DNA looping in other words, different interactions between promoters and enhancers, affecting gene expression.

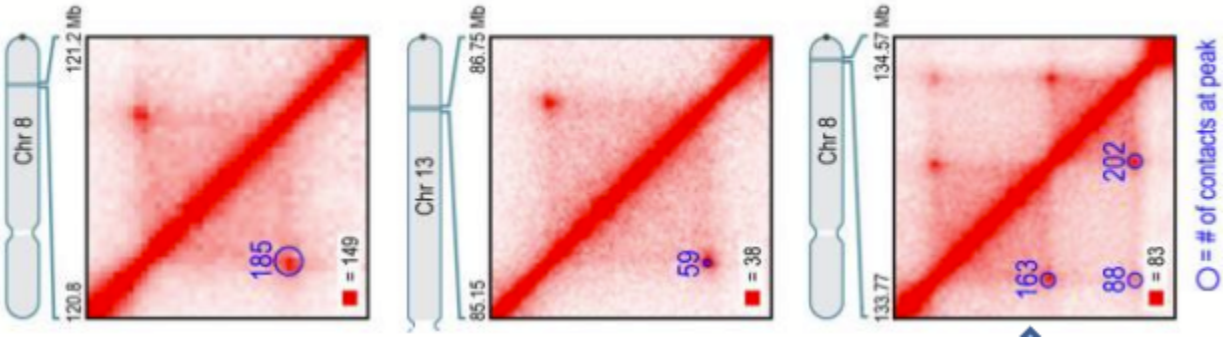
Methods to model chromatin looping

- Hi-C: models long-range DNA2DNA interactions over entire genome



- This outputs this Hi-C mapping we see above, where the green dots show an example of areas in the genome which are predicted to be very close to each other, although they are far in the genome

On the TAD Level this looks like



- ChIA-PET, using a protein of interest (CTCF, YY1,...) then provides information about chromatin interactions that are specifically associated with the protein targeted by the antibody used. This allows researchers to understand how certain proteins (like transcription factors or histones) influence the structure and function of the genome

Both are used to train CNN, RNN or transformer models to predict chromatin folding

- Input DNA sequence only, Output: prediction of interaction score for pairs of genome regions

DeepC CNN based approach:

- Add dilated layers to widen receptive field
- Important idea 1: Remove domination of close interactions, transform row counts to quantile based scores -> predicts skeleton of interactions
- Important idea 2: (Pretrain) Use transfer learning, transfer weights from training on chromatin data, which should encode DNA “grammar” and then use it as input to predict the chromatin interaction scores
- DeepC allows predicting changes in 3D DNA interactions due to mutations and deletions
- Drawback: One must train in a cell-type specific way, so extrapolation is not possible

DeepTACT RNN/LSTM based approach:

- Does not consider DNA sequence in between the two elements
- Uses only one hot encoded promoter and enhancer sequence and chromatin accessibility score (DNase-seq)
- Uses conv layer before LSTM module

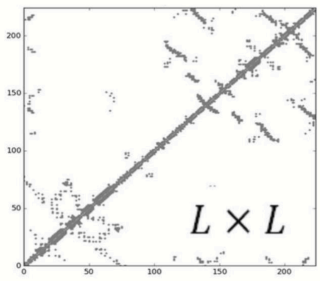
TransEPI: Transformer based architecture

- Conv layers + Max pool succeeded by Transformer module
- Self-attention, sometimes called intra-attention is an attention mechanism relating different positions of an input sequence in order to compute a representation of the sequence
- Multi-Head Attention: allows attention function to extract information from different representation subspaces
- Positional encoding
- Chromatin features (CTCF-binding sites, chromatin acc, histone marks) used in input

Prediction of 3D structure of proteins

- Alphafold
- MSA: For protein MSAs, each row represents the amino acid sequence of a protein from different species or different protein variants within the same species.
    - By comparing the co-movement of amino acids across species, important sequences can be identified. This is crucial in the AlphaFold process as they provide evolutionary information that is key for predicting the 3D structures of proteins.
- RaptorX: CNN-based framework to predict protein contact maps
    - predicts the spatial relationships between amino acids in a protein.
    - **Input**: Protein sequence, Evolutionary information from MSA
    - **Output**: Contact Maps, i.e., predicts which amino acids are close in 3D space, and Distance maps, i.e., estimates distances between residue pairs.
    - **Architecture**: 1D convolutions capture sequential information, while 2D convolutions capture pairwise residue features. Uses a 2D CNN to process pairwise residue features. Combines 1D sequence features (e.g., evolutionary data) with 2D spatial features to predict protein folding.
    - Output: contact map, is a 2D representation of the folding, where the points on the graph show positions in the protein which are predicted to be close in 3D space.

Output: Contact map



LECTURE 6: Deconvolution of mixed transcriptomics signals

- Bulk RNA-seq protocol: Fragment DNA, adapter ligation, and then amplify using PCR finally sequence, map to reference genome and quantify expression
- Bulk RNA-seq profiles of transcriptome of entire experimental sample
    - Take entire tissue sample and sequence an average signal
- Single cell RNA-seq
    - Identify cells on single-cell level and profile individually
- Using bulk RNA of a tumor could identify only predominant cell type and ignore treatments for sparsely available cells
    - Remedy? Sc analysis or signal deconvolution
- Standardisation:
    - First Raw read count which included individual nominal reads of each gene
    - This is normalized to account for gene length and library size (i.e. the total number of reads)

Signal deconvolution: Separate the noisy bar sounds into their individual sources (cocktail party)

- Allow pseudo-single cell analysis of bulk RNAseq data
- Specific composition of Tumor microenvironment is very informative for health outcomes.
- Bulkseq is used as it can be stored for long and it is much cheaper than scRNA-seq

Blind Signal Deconvolution X (genes * samples) = S (cell types) * P^T (proportions)
Independent Component Analysis (ICA):

- Write X \sim A* S
    - A is a mixing matrix which includes the pseudoinverse of the weight matrix
    - S is the “source” matrix which includes the cell types, where the sources are assumed to be independent and non-gaussian
- FastICA is an efficient implementation for ICA
    - It defines an unmixing matrix W s.t. It maximizes non-gaussianity
    - ICA cannot be used on a mixture of gaussians
    - Only 1 gaussian source is acceptable
    - Maximizing non-gaussianity is equivalent to maximizing independence (CLT as sum of RV tends towards Gaussian, so my maximizing non-gaussianity we’re maximizing independence)
    - Negentropy (negative entropy) is used to optimise as: Gaussian RV has largest entropy among all RV with equal variance , so negentropy “measures” the distance to normality
    - Preprocessing: Centering and Whitening which transforms the input data into a new set of variables that are uncorrelated and have unit variance
- ICA is NOT deterministic as it begins with random init of weights
- What should k be? Method is not deterministic
- To deal with non-determinance of ICA, use parallel consensus ICA where fastICA is run 1000 times separately and consensus S and A matrices are found by averaging all reordered S and A matrices.
- However S and W can be negative, so

→ Non-negative matrix factorization (NMF)

- Find W,H which minimize 0.5*||X-WH||_F^2

- Euclidian update
- NMF may be better for DNA signal deconvolution

Supervised / guided signal deconvolution

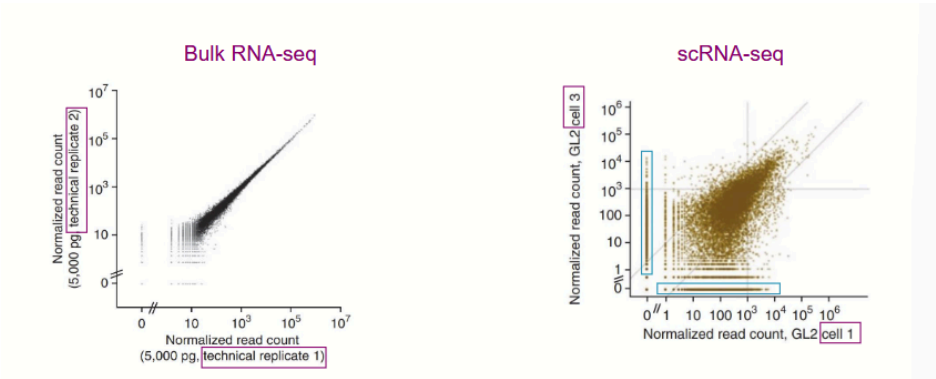
- Input: A reference gene expression matrix for cell types of interest
- OR a set of marker genes
- SO: will only look for cell types provided in reference
- CIBERSORT: Uses reference matrix and combines it with Support vector regression to estimate composition
- m = f (weights of cell type) x B (gene expression signature matrix)
- In CIBERSORT, the support vectors represent genes selected from the signature matrix and the orientation of the regression hyperplane determines the estimated cell type proportions
- Performance metrics: Pearsons corr, mean absolute deviance
- Alternative linear least-square regression
- Reference based perform better

LECTURE 7: single cell transcriptomics and dropout imputation

- Same procedure like bulk RNA-seq but add a cell-specific barcode to each genome fragment
- In 10x genomics, barcode is added using gel beads coated with a unique oligo sequence
- Number of single cells that one can profile in one experiment grows ~exponentially
- UMI (Unique molecular identifier) allows to identify PCR duplicates, however we want to sequence unique UMIs as not to bias the data
- → Reads with unique UMIs are counted, low quality cells and silent genes filtered

Dropout effect:

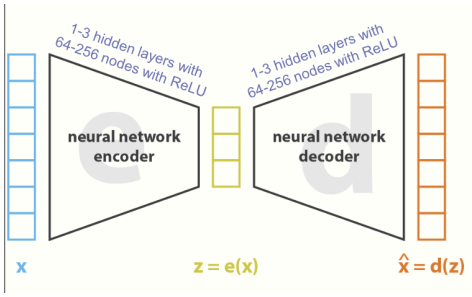
- Not all transcripts are captured in each cell, so some replicated could have a count of 0. In other words, some expression groups are not captured at all.
- → scRNA-seq: Zero-inflated distribution



Visualizing scRNA-seq

- PCA: Linear transformation of the data to inspect presence of technical and/or biological biases
- t-SNE (t-distributed stochastic neighbor embedding) and UMAP (uniform manifold approximation and projection) are non-linear transformations to visualize clusters
- t-SNE and UMAP work by matching the pairwise similarities of the low and high dimensional representations of the data
- We define a high and low dimensional joint probability
- In t-SNE we minimize the KL(p||q) and in UMAP we minimize the Cross entropy between the two distributions
- t-SNE cost function is not convex, so result depends on initialization
- t-SNE and UMAP give similar visualizations, but UMAP is much quicker to run
- UMAP can use any distance metric, not only euclidean

Using Autoencoder architecture to find biologically meaningful latent embeddings



- Where the loss is the ||x - \hat{x}||^2
- In Variational Autoencoders, in order to regularize, assume latent space is parametrized by a normal distribution with mean and variance
- Then sample from this distribution and use this as an input to the decoder
- Loss VAE: ||x-\hat{x}||^2 + KL(N(\mu,\sigma^2),N(0,I)||
- The embedding space of the AEs can be used for visualization with t-SNE and UMAP

ZINB:

- Due to dropout phenomenon, zero-inflated NB distribution of read counts is added to loss

Conditional VAE:

- Information on variables we want to "control for" or have minimal effect on latent space, the conditions are appended to both the input X and the latent space Z. The model therefore, does not model the condition
    - Cell cycle effects e.g.
- A penalty can be added to the loss function to ensure that the latent space does not capture the conditions (MMD penalty)
- ZINB VAE and ZINB CVAE allow to estimate 'true' transcript expression and therefore impute dropout effects

Imputation of scRNA-seq data

- How to measure accuracy of imputation? Correlation between bulk RNA-seq and bulkified scRNA-seq data
- **Imputation with diffusion: Markov affinity-based graph imputation of cells**
    1. Transform read counts to remove library size bias and gene size bias
    2. Apply PCA on transformed read counts K=[20,100]
    3. Calculate distances in PCA-space
    4. Compute an affinity matrix A based on step three using an adaptive gaussian kernel
    5. Perform a row-stochastic normalization rendering A into a Markov transition matrix M
    6. Exponentiate M to M^t to represent probabilities of random walk with length t
    7. Then X_{imputed} = M^t*X
    8. X_{imputed} is finally rescaled to account for the decrease in non-zero matrix entries which were previously empty
- Diffusion time t describes the amount of smoothing that the data undergoes
- Zero counts can therefore theoretically be imputed using information from similar cells

LECTURE 8 - Batch correction, Clustering and differential gene expression and cell type annotation in scRNA-seq

- Batch effect: technical variability arising from combining data from different experiments where technical variability arises, such that the variability is only technical
    - e.g. lab conditions, personnel, time of day, instruments
- Regression based methods for batch correction: Linear models
    - Y ~ Batch effect (basically different intercepts per batch)
    - Linear Mixed models, fixed effect of gene, random effect of individual and batch

        - Bayesian framework: assume expression Y_{gene,individual,j:batch} ~NB(\mu_{gij},\phi_{gi})
        - \phi_{gi} is the dispersion batch effect and \gamma_{gi} is the batch effect
        - Parameters are estimated using edgeR and then \mu* is found by subtracting \hat{gamma} and \phi* = mean(\hat{\phi})
        - The data is finally adjusted b mapping to the estimated batch free distribution ~NB(\mu*,\phi*)
- Dimensionality reduction based methods
- Canonical Correlation Analysis (CCA)
    - Find projections u and v of matrices X and Y which maximizes the correlation between the projections
- Integrative Non-negative Matrix Factorization (iNMF)
    - Reconstruct dataset E_i with lower dimensional matrices H_i, W and V_i such that E_i \sim H_i
- Harmony: major assumption is that the cell type separates observations more than experiments
    - Using Maximum Diversity Clustering on PCA embedding, assign soft cluster assignments to cells.
    - MDC is an extension of soft K-means clustering which maximizes batch-diversity within a cluster
    - By subtracting all batch specific elements of the embedding Z, we output a batch-regularized embeddings
- CVAE: remember loss ||x-\hat{x}|| + KL(N(0,I),N(\mu,\sigma^2))
- Transformer based archs like scGPT
- Generally deep learning and dimensionality reduction based techniques perform well
- Applying K-means on high-dimensional space wouldn't work due to curse of dimensionality
- Also: rare cell types (differently size clusters), large number of cells, noisy (+ dropout), batch effects + cell-cycle effects

Clustering:

- True labels based architecture: K-means based, Hierarchical clustering based, community detection based (Leiden, louvain)
    - Louvain clustering finds communities within a graph optimising for modularity, which optimises inter and intra-community variance
    - Leiden clustering adds a refinement step to prevent **formation of Disconnected Communities**
- Cell types represent functions of a cell and are invariant
- Cell states represent more labile current situations, making them often harder to annotate

differential gene expression analysis (DGEX)

- Using differential gene expression analysis (DGEX), we can find genes characteristic of cell types or states
- Heuristically, we will cluster the gene expression in some embedding in order to create groupings which describe the cell type e.g

Gene Set Enrichment Analysis (GSEA)

- Is used to find links between a set of differentially expressed genes and known pathways
- a "pathway" refers to a series of actions or interactions among molecules in a cell that lead to a certain product or a change in the cell.
- Using marker genes: a library of genes associated with certain cell types and prior knowledge, GSEA is applied allows to incorporate prior knowledge of pathways/co-expression
    1. Compute the Enrichment Score (ES)
        a. quantifies the degree to which a predefined set of genes is overrepresented at the top or bottom of a ranked list of all genes in the dataset.
        b. We do this by using the Kolgomorov-Smirnov (KS) like Random Walk statistic
        c. ES(S) is the maximum deviation from zero of the RW
    2. Estimate significance level
        a. Empirical phenotype permutation test: By randomly permuting labels and recomputing ES to find distributions, then find significance level
    3. Normalize ES to get NES
    4. Adjust for Multiple testing
- Cell type annotation is based on marker or reference
    - In both cases, one needs good marker genes or reference (which is strong enough to account for batch effects)
- Annotating single cells vs clusters: Annotating single cells is a bit more variable
    - Annotating clusters leads to badly clustered cells being annotated incorrectly, additionally dependence on clustering tech

Marker-based method: GSVA

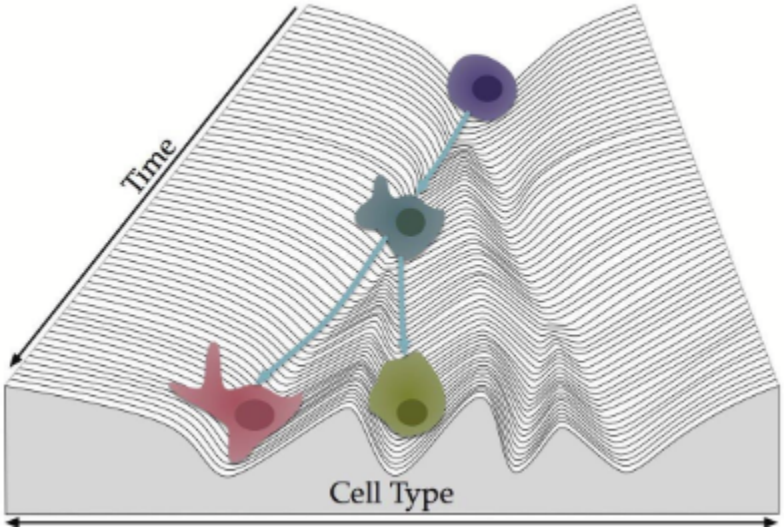
- Input: Gene expression X (normalized), Output: Scoring of genes sets
    1. Use a kernel estimation of CDF(x) to correct for bias in RNA-seq
    2. Transform to ranks and make ranks symmetric around zero to reduce influence of outliers
    3. Then same approach like GSEA, find KS like RW stat, and normalize

Reference-based method: scmap

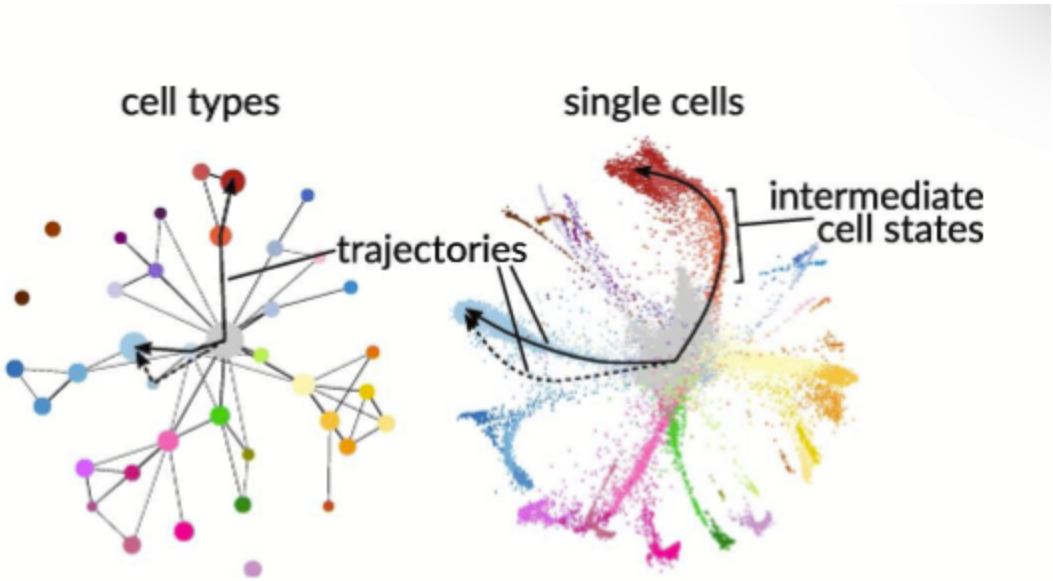
- Represent each cell by a set of sub-centroids found with K-means based on only a subset of features
- Compute nearest neighbors using sum of distances to sub-centroid
- Assign to cell to type using nearest neighbor
- Mark cell unassigned if (a) three NN do not belong to same cell type in reference (v) the cosine similarity between the cell and its nearest neighbor is <0.5

LECTURE 9 - Trajectory analysis from scRNA-seq

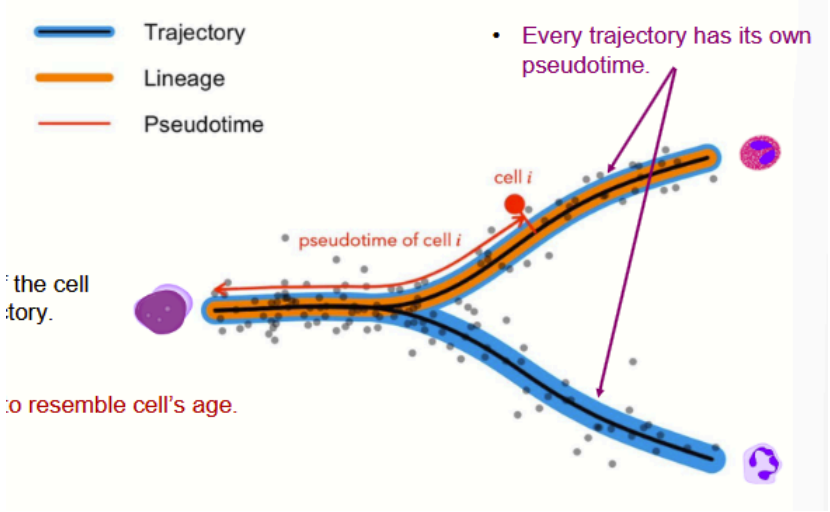
- One can think of cell differentiation as a rugged topology over time, where depending on the presence of TFs or lack thereof, cells differentiate along different topologies “Waddington landscape”



- Transdifferentiation and dedifferentiation can happen as a reaction to stimuli
- Yamanaka Cocktail describes a set of 4 TFs, coined “reprogramming TF” which **reprogram mature cells to pluripotent stem cells**, in other words can be differentiated to many cell types once again
- Another example is using a set of three TF, on can reprogram tumor cells to dendritic cells *in vivo*
- Takeaway: Assume we can describe the cell type landscape with a graph. Then one could aim to find trajectories along this graph describing the developmental trajectories



- When building developmental trajectories, we map to pseudotime, which is a measure of the cell development along its trajectory, which does not necessarily need to resemble cell age



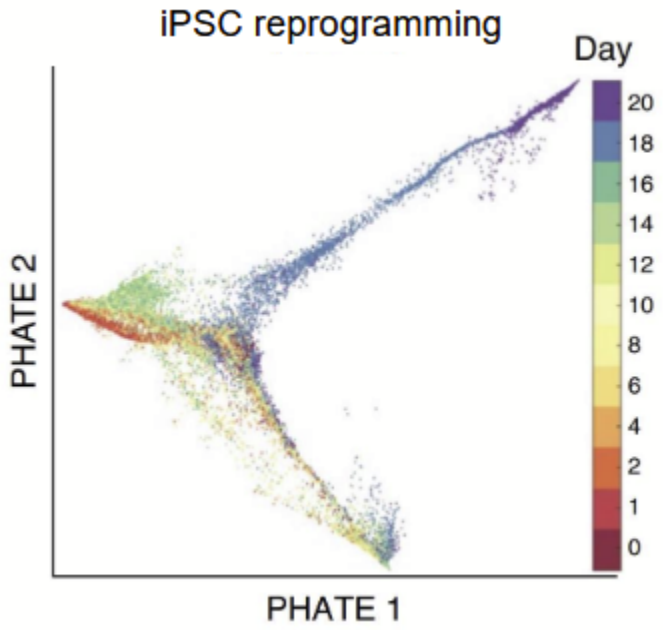
- In essence, one would like to find trajectories of cellular development along pseudotime. How?

Manifold-learning approaches:

- **PHATE**: a visualization method that captures both local and global nonlinear structure
  - Essentially, we want a method that captures development along time, while still capturing the clusters/grouping in the differentiation
  - PCA seems to find time-progression and UMAP and t-SNE find the clusters
  - PHATE seems to do a good job finding both
  - In general, one finds a lower dimensional manifold with a dimensionality much less than 20K. This relationship is described by a nonlinear embedding

- Additionally, we replace the Euclidean distances in the high-dim space with informational distances which are then embedded
- PHATE uses the distance function to measure similarity, by using a locally adapted gaussian kernel
- Local affinities are transformed to diffusion operator to be use in a t-step random walk, which is hoped to approximate the embeddings of the geodesics on the latent manifold (similar to MAGIC)
- The diffused affinities are transformed (log transformed transition probabilities) to new distances
- Embed distances in low-dimensional space using metric MDS minimizing the stress function

- Output:



- In my understanding, PHATE seems to capture trajectories well and not hallucinate trajectories when there are none
- It preserves a range of patterns in data, including continual progressions, branches and clusters
- PHATE is applicable to a wide variety of data types, including mass cytometry, scRNA-seq data, Hi-C and gut microbiome data

Monocle:

- A popular, graph based method for trajectory inference in scRNA-seq field

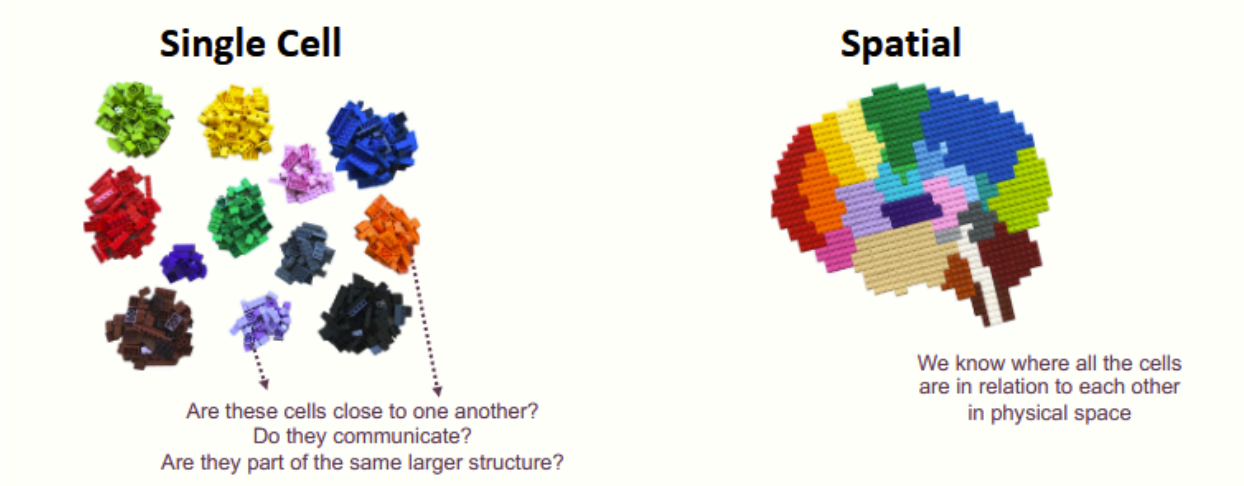
- Using a minimum spanning tree on the data projected into an embedding
    - MST: Subset of edges of a connected, edge-weighted undirected acyclic graph which connects all the vertices together, without cycles minimising the total edge weight
    - Kruskal's algorithm: to build an MST
        - Add the next lowest-weight edge that will not form a cycle to the MST
        - Complexity O(ElogV) E: number edges and V number vertices
- ICA is used to project to the embedding
- Cells are nodes, and the edge weight is the euclidean distances between the ICA representations
- The backbone is the tree diameter

Monocle 2: Uses Reversed graph embedding (RGE) to better determine the branching structure using principal graph

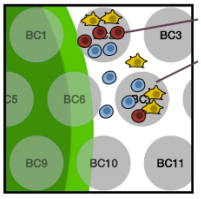
- Underlying hypothesis: cells of the same type should form dense clusters in a latent space, where these clusters should be connected with paths representing intermediate cell states
- RGE aims to learn both a set of latent points and an undirected graph connecting these points
- RGE simultaneously learns a principal graph representing the cell trajectory, as well as a function which maps cells to these trajectories back to the original high-dim space.
- In essence,
    1. Apply dimensionality reduction on input X to get Z
    2. Guess initial cell trajectory on centroids
    3. Update cell positions based on trajectory
    4. Update cell centroids
    5. Repeat
    6. Update map to high-dim space
    7. After convergence, select root and find pseudo time along branches
- Monocle 2 performs much better!
- Benchmarking studies show that the specific topology matters to the method performance

LECTURE 10: Spatial-omics

- We already know about measuring gene expression using bulk and scRNA-seq approaches
- Spatial omics add the location coordinates to the scRNA-seq data, to better inform the gene expression using a spatial context
- Especially as in many manifold learning techniques, we the distances in the latent space is not always informative, this could be paramount

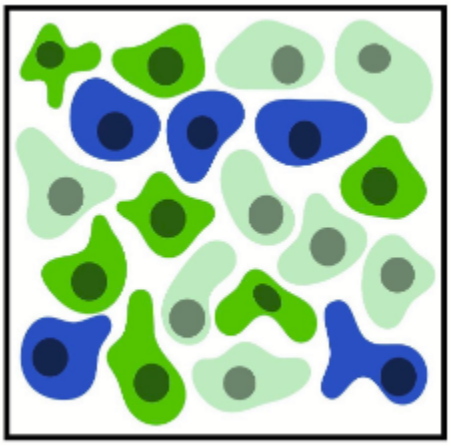


- Early spatial methods, used simply histological images, or just images from the microscope
- Spatial profiling technologies by scale
    1. Multi-cell resolution: All transcripts in a predefined area are measured. We receive aggregated expression of all cells in captured area



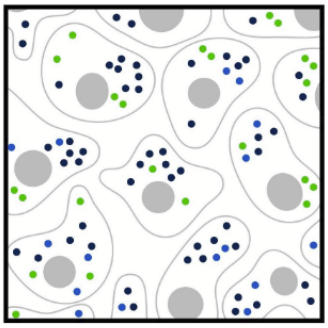
- Cells that fall outside spots are NOT measured
- Coverage usually transcriptome wide
- 10X Visium and slide-seq

2. Single cell resolution and spatial profiling



- Image molecular features of single cells at their exact position (EXACT BUT SUBSET OF TRANSCRIPTOME)
    - Measure limited feature space can be expensive
- Capture spots on the scale of single-cell (Visium HD or slide-seqV2) (CAPTURE FULL TRANS., LESS EFFICIENT)
    - Measure full transcriptome, but lower capture efficiency
    - Untargeted experiments - get full RNA space

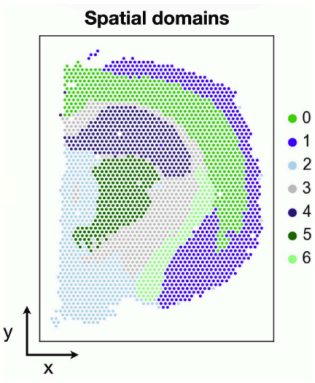
3. Sub-cellular resolution spatial profiling



- Capture individual molecules, through single-molecule imaging
    - Xenium, MERFISH
    - Segment individual cells from sub-cellular data

Spatial data analysis approaches:

- Spatial domain analysis: once again very high dimensionality
    - We have dimensionality of upto ~ 20K for gene expression and 2D for spatial
    - Possibly add histology
    - Use both gene expression and spatial dimension to cluster and output



- This results in the spatial “domains” visible on the left
- Spatial domains describe different anatomical structures

- Louvain clustering on scRNA-seq data, can now be extended in spatial omics data by adding spatial dimensions

SpaGCN:

- Integrates gene expression, spatial location and histology to identify spatial domains
- SpaGCN uses additional z dimension, representing histological similarity
- SpaGCN is based on building a graph G representing spatial proximity
- Vertices in graph are connected with undirected weighted edges
- Edge weights measure similarity between spots using a distance metric
- Graph structure stored in NxN adjacency matrix A

Graph Neural Networks:

- In each layer, information from each node is passed on to neighboring nodes
- The current feature vector is updated using information from itself in the last time t and weighted information from adjacent feature nodes
- f(X,A) = \delta =RELU(A=adj matrix, X = Features, B = Weights)
- The filter parameters in B are shared across all vertices in the graph and are automatically updated during an iterative training progress
- Iterative clustering to identify spatial domains
    1. Initialize clusters with leiden clustering in gene space
    2. Define q_{ij} = distance between spot i and centroid j = probability of assigning spot i to cluster j
    3. Define p_{ij} = auxiliary target distribution
    4. Train GCN parameters and cluster centroids to minimize KL(p||q)

Spatial mapping of scRNA-seq data:

- We know from above, that scRNA-seq and spot-resolution spatial data is fully resolved in gene space
- No current spatial method combines this information (except for Viium HD, which is expensive and has low capture rate)
- AIM: map scRNA-seq data to corresponding location using Visium as a guide.
- Output: scRNA-seq fully resolved in gene space with spatial map

scDOT: optimal transport for mapping senescent cells in spatial transcriptomics

- Input scRNA-seq and spatial transcriptomics (multi-cell resolution, fully resolved)
    - Spatial transcriptomics: the multi-cell “spots” are deconvolved into spots and cell types
    - Using optimal transport the scRNA-seq data is then mapped to the spots on the resolution of the convolved cell type matrix
- Transport theory: seeks for best way to ship stones, minimizing transportation cost

- In math, transport refers to ways in which one describes the transformation of one point cloud to another
- In genomics, different assays (Hi-C, single cell RNA-seq, ATAC-seq (open chromatin)) are destructive, so cannot be repeated on same cells
- In order to reconstruct, the full picture of the cell population, we need to align or map the two separate sets of cells (from Hi-C, scRNA-seq & ATAC-seq experiments)
- Nearest neighbor matching: most simple approach, where we match points to a nearest neighbor, some points could stay unmatched and some are selected repeatedly
- One-to-one matching
    - Assume Set Y is some permutation or set X (Cannot be used for different sized sets)
    - And then find a map of the correspondence of points in X & Y
- Transportation plans:
    - Instead of permutation: construct a transport plan using a coupling matrix P_{ij}
    - P_{ij} measures the association between cell x_i and y_j
    - U(a,b) describes the set of coupling matrices which conserve mass
    - Define Cost function: weighted sum of distances between matched points
    - OT(a,b) min Cost
- scDOT: Optimal transport informed by deconvolution (P = cell type proportion matrix)
    - Y = spatial transcriptomics profile of dimension m=spots*p=genes
    - P = Cell type proportions (to be estimated) of dim m*c = cell types
    - Estimate Y by PS
    - S = Signature matrix (known gene expression profile for cell types) m*c
    - Find P* = argmin ||Y-PS||_F can be solved using non-negative LS
    - \gamma is transport plan
    - X = scRNA-seq data n*p, Y spatial m*p
    - M = cost matrix defining distance between cells of X and spots of Y
        - We define the distance using a cosine distance (scale invariant, account for measurement sensitivities of technologies)
        - Find optimal transport plan by minimizing cost and applying an entropy regularization

LECTURE 11: Integration of different data-types in sc experiments

- Data Modalities aka multi-omics are measurements and analysis of multiple omics data together
- Multi-omics allows more comprehensive insights into cell functioning and more successful downstream analysis
- Especially when one modality does not describe all information, multi-omics allows to impute missing modalities and translating between modalities.
- Matched multi-omics data are from the same cells, unmatched not
- CITE-seq: Simultaneous sequencing of scRNA-seq and surface protein abundance
    - Allows multi-omics analysis: proteinome + transcriptome
- Early, middle and late integration of multi-omics data
- SEURAT: matched use weighted NN
    - Produce a weighted similarity graph for each modality (kNN)
    - Integrate them to clearly a weighted similarity graph (weight retrieved through assessment of predictive power of resp. modality)
- SEURAT Unmatched: CCA, MNN and anchors
    - Assume common underlying cell population
    - Requirement for Seurat: same dimensionality of the integrated modalities
    - If modalities are identical, data integration with Seurat is equivalent to batch effect correction

Approach:

1. Data prep and feature selection
2. Dimension reduction (CCA) and identification of anchor correspondences between datasets with mutual NN

→ Anchors are two cells (from each dataset), that are predicted to originate from a common biological state. All other cells without corresponding perfect matches, will then be aligned across modalities based on the anchors.

3. Filtering, scoring, weighting of anchor correspondences
4. Data matrix correction
   → For a Batch/Modality effect B =Y-X, we create a matrix C = BW^T such that one removes the effect using \hat{Y}=Y-C

Deep Learning / Autoencoder based cross-modality integration:

- Create cross-modal autoencoders, which for modality-specific encoder-decoders, one creates a shared latent space. This allows translating between modalities at scRNA-seq level
- Latent space representation of cell can then be used for downstream task
- This is done with LIKE IN SPATIALGLUE, with a reconstruction loss and a divergence loss. The first of which makes sure the modality specific \hat{x} of x does not diverge and the other, which makes sure the shared latent space is there

ENVI: Environmental Variational Inference: embedding spatial and scRNA-seq data into latent space.
It uses a **conditional VAE** that jointly embeds spatial and scRNA-seq data into a latent space.

Applications: Enhancing spatial transcriptomics data resolution; providing insights into cell-cell interactions and tissue organization; supporting biological discoveries by connecting spatial context to gene expression profiles.

The main point of ENVI is to integrate **spatial transcriptomics (e.g., Xenium or similar technologies)** and **single-cell RNA-seq** data into a **shared latent space**, enabling fine-grained cell type annotations, inference of spatial niches, and imputation of missing data while accounting for spatial and molecular variability.
→ It infers missing data or low-coverage gene expression from spatial data by leveraging the rich molecular information in scRNA-seq, which is particularly useful for limited-marker spatial datasets.

ENVI uses a CVAE, where encoders and decoders handle modality-specific data (spatial or molecular), and the shared latent space allows integration, which enables tasks like clustering, visualization, and prediction across modalities.
It uses a **composite loss** including:
- Reconstruction loss (ensuring faithful recovery of input data)
- Divergence loss (aligning distributions in the latent space)
- Spatial and molecular consistency loss (cross-modal consistency), which helps align spatial and molecular modalities in the latent space

LECTURE 12: Survival Analysis

- Data (right-)censoring: Censoring occurs when a sample does not “observe” the event before the end of the study, so either the patient does not follow up, withdraws, or they never experience the event during the study
- Survival function S(t) = P(T>t), probability to survive longer than *t*
- Kaplan-Meier estimate \hat{S}(t):

$$\hat{S}(t) = \prod_{t_i<t} \frac{n_i - d_i}{n_i} \quad \text{or} \quad \hat{S}(t_i) = \hat{S}(t_{i-1}) * \left(1 - \frac{d_i}{n_i}\right)$$

Where,

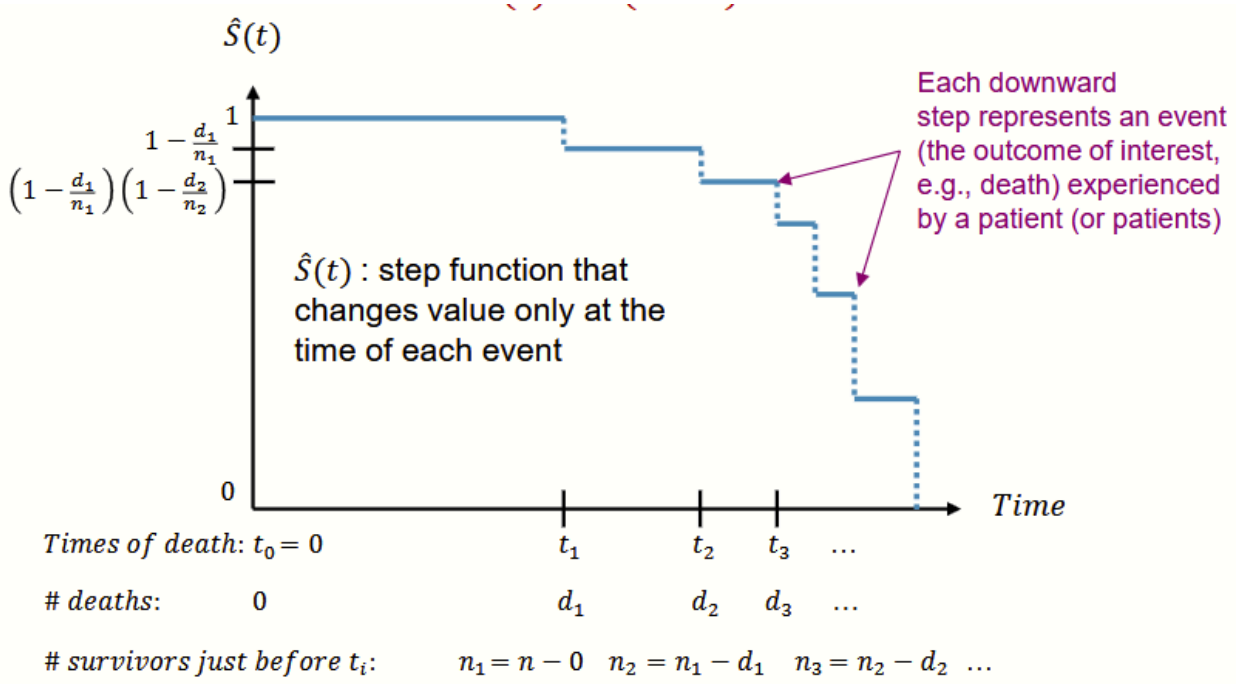
$\hat{S}(t_{i-1})$ = the estimated probability of being alive at t_{i-1}

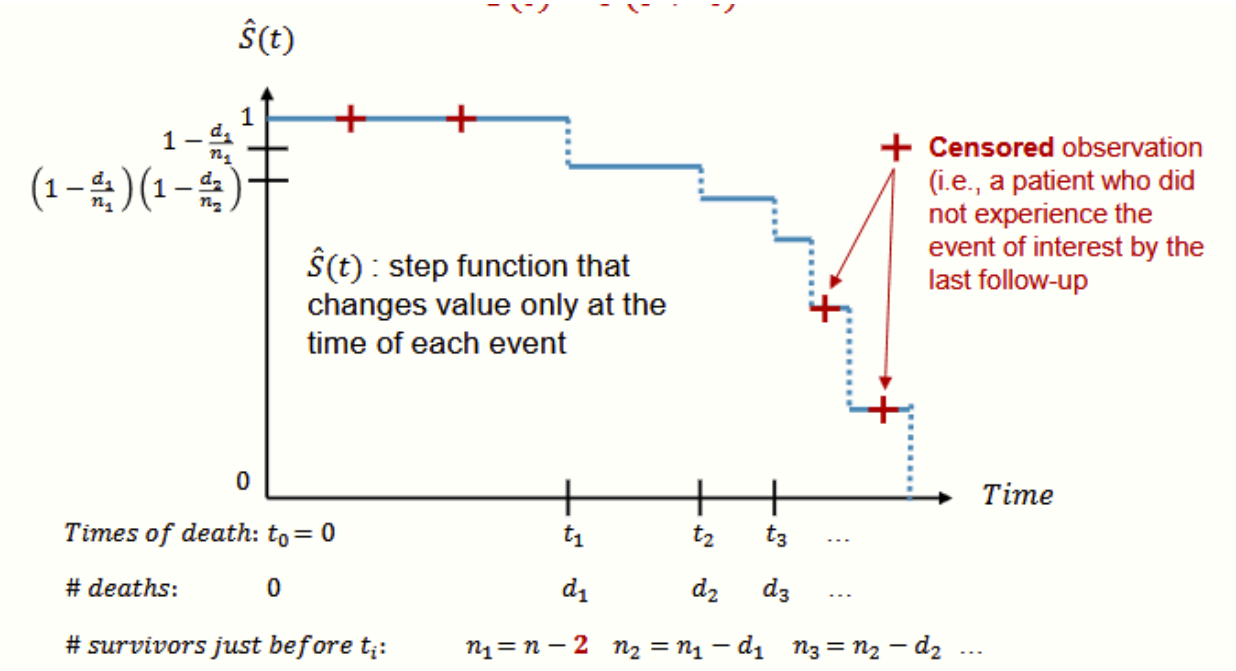
n_i = the number of patients alive just before t_i

d_i = the number of events at t_i

t_0 = 0

$S(0)$ = 1





- Overall Survival (OS), the time tot death
- Relapse free survival time, corresponding to time between response to treatment and recurrence of disease (EFS)
- Significant difference in survival among groups? Log-rank test

H_0: There is no difference regarding survival among two groups

We look at two groups of patients (treatment vs. control)
Let $1, \dots, J$ be distinct times of observed events in either group
Let $n_{1,j}$ and $n_{2,j}$ be the number of subjects "at risk" (who have not yet had an event or been censored) at the start of period j in the groups, respectively
Let $d_{1,j}$ and $d_{2,j}$ be the observed number of events in the groups at time j
Define $n_j = n_{1,j} + n_{2,j}$ and $d_j = d_{1,j} + d_{2,j}$

Under H_0, for each group i, the observed number of events $d_{i,j}$ at time j follows a hypergeometric distribution with parameters $n_j, n_{i,j}, d_j$.
This distribution has expected value $E_{i,j} = n_{i,j}\frac{d_j}{n_j}$ and variance $V_{i,j} = \frac{E_{i,j}(n_i - d_j)(n_j - n_{i,j})}{n_j(n_j - 1)}$.
The log rank statistic now compares $d_{i,j}$ to its expectation $E_{i,j}$ under H_0:

$$Z_i = \frac{\sum_{j=1}^{J}(d_{i,j} - E_{i,j})}{\sqrt{\sum_{j=1}^{J} V_{i,j}}} \to N(0,1), \quad (i = 1 \; or \; 2)$$

The test statistic is then $Z_1^2 + Z_2^2$. This follows a Chi-Squared distribution with m-1 degrees of freedom, where m is the number of groups.
$Z_1^2 + Z_2^2 \to \chi^2_{m-1}$

→ We can calculate the p-value!

- Maximum Power when assumption of proportional hazards is true

Cox Proportional hazards model:

- Remember h(t): hazard is the instantaneous rate at which an event occurs
- Cumulative hazard is \int_0^t h(u)du
- Estimate is \sum d_i/n_i
- Cox model : h(t|X) = ho(t)exp(X\beta)
- exp(beta_i) is called HazardRatio, beta_i = log-hazard-ratio HR>1 increase in hazard

COX prop hazards assumption:

1. Relationship between X and log hazard is linear
2. In absence of interactions, predictors act additively on log hazard
3. The effect of predictors is the same for all values of t

Interpretation of Cox hazards

- Increasing x by 1 increases hazard of event by exp(beta_j) AT ALL POINTS in time
- Ratio of hazards between person x and x* is exp((x*-x)\beta)
- Model makes no assumption about form of h0(t)
- After fitting model, we know their relative risk ratios, and we can rank the patients by their risk
- Full survival probabilities are estimated from model

Fitting Cox PH model:

- Maximise partial likelihood L(beta)

Under the Cox PH assumption, this is

$$L(\boldsymbol{\beta}) = \prod_{i:C_i=1} \frac{\exp(\boldsymbol{X_i\beta})}{\sum_{j:Y_j \geq Y_i} \exp(\boldsymbol{X_j\beta})}$$

Taking the logarithm, we get

$$logL(\boldsymbol{\beta}) = \sum_{i:C_i=1} [\log(\exp(\boldsymbol{X_i\beta}) - \log \sum_{j:Y_j \geq Y_i} \exp(\boldsymbol{X_j\beta})]$$

Thus, the loss function minimized by the Cox model is the negative log likelihood:

$$\text{Cox PH loss function} = -logL(\boldsymbol{\beta})$$

And we can use gradient descent to minimize it.

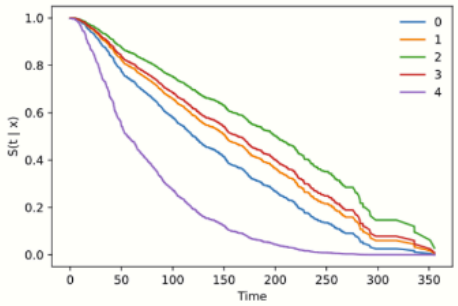
- Can add regularisation to make Lasso Cox : -logL(beta) + \lambda ||\beta||_1

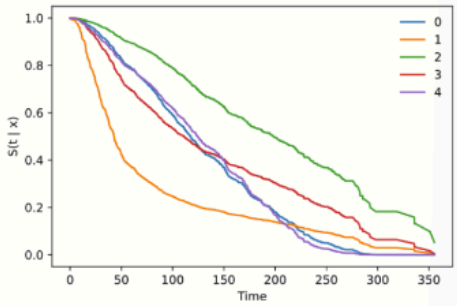
Quality of fit of Cox PH model:

- Concordance index: = #concordant pairs /(#concordant + #discordant)
- C-index = 0.5 for random predictions

Relaxing assumptions of Cox PH model:

- DeepSurv: Relax log-linearity of hazard, replace with function g, paramterized using neural-net (non-additive and nonlinear)
- CoxTime: Relax assumptions of time-independence: Enables time-dependent effects of covariates. Model h(t) = h0(t)exp(g(X,t))





For the two survival curves above: The left one is DeepSurv and the right one is CoxTime. → **The estimated survival curves can only overlap if we include time as a covariate, otherwise they cannot!**
Without time as a covariate, the model assumes proportional hazards (relative risks stay constant over time). This fixed relationship prevents survival curves from crossing.

ADDING MULTI-OMICS: How to deal with multi-modal data integration problem

- Early Integration: Group Lasso with m omics groups, group-sparsity
  - Sparse Group Lasso: Sparsity within groups too
- Middle Integration: MultiSurv
- Late integration: priorityLasso

→ Tries to always explain the yet unexplaiend variance which was not explained by the previous modalities! → Works really well.