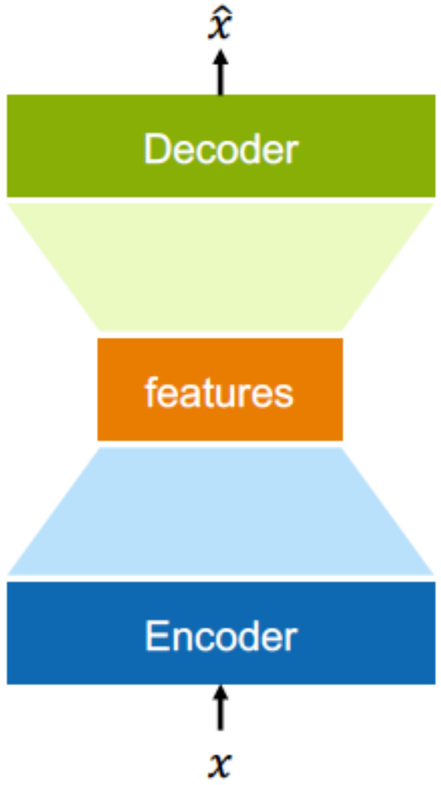
Maximum Likelihood Estimators and The Expectation Maximization Algorithm

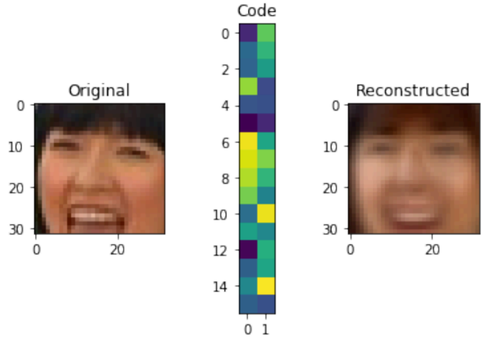
LALEH HAGHVERDI

Autoencoders

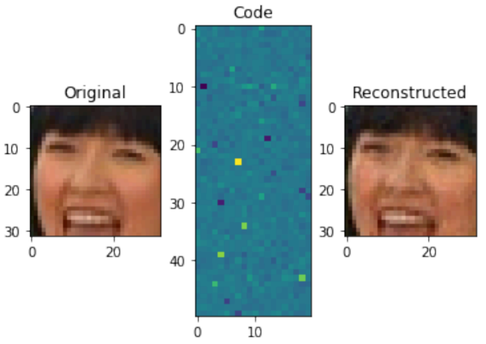


- **Unsupervisedly** learn condensed representation of data through autoencoding task
  - Encode the input into lower-dimensional latent features
  - These features should allow reconstruction of the input
  - Optimize model to minimize reconstruction loss, e.g.

$$L(x, \hat{x}) = \|x - \hat{x}\|^2$$

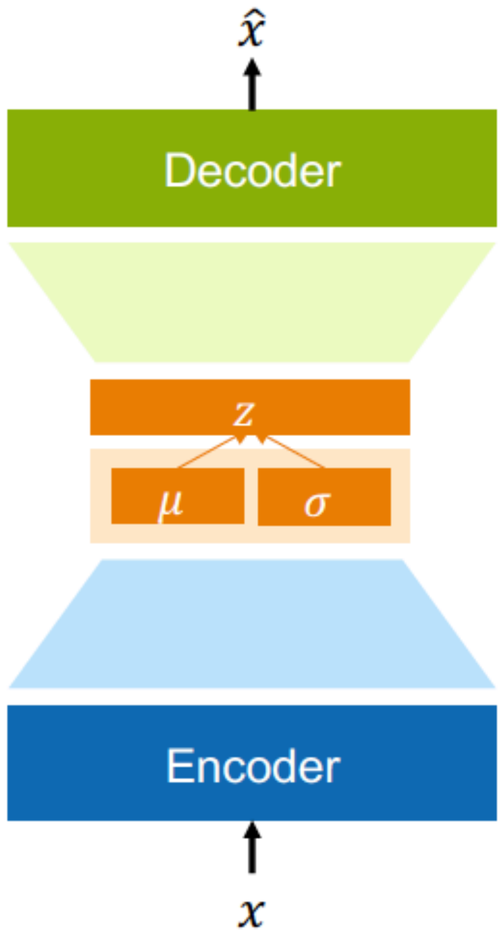


- AE gives features for reconstructing the data
  - The bottleneck forces the model to learn rich important features of the input by ignoring noise in the data
  - However, mapping between input and features are deterministic
    - Feature extraction
  - Can we modify the model such that we can generate more data from it?



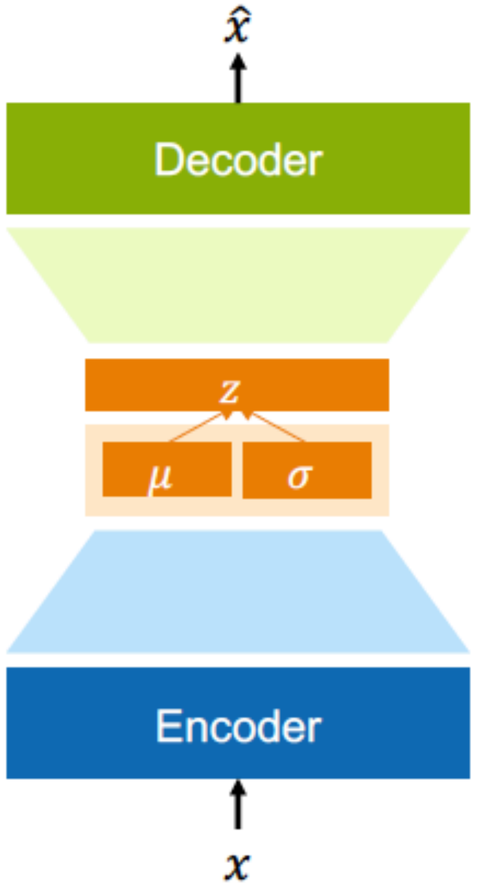
- Can be used as a DR method, eg. scVI
- Data compression and feature extraction

Variational autoencoders



- Instead of deterministic mapping, VAE models the **distribution** of the latent variables

Variational autoencoders



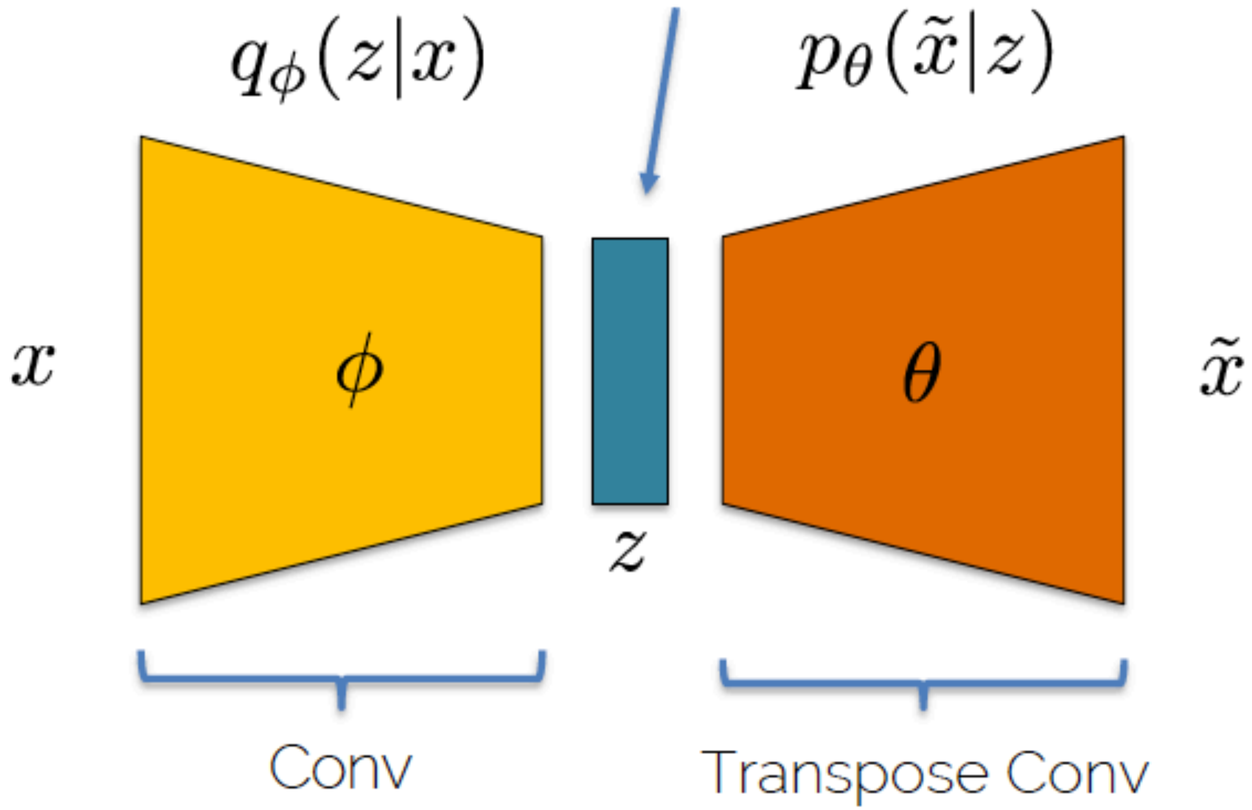
- Decoder generates new data conditioned on z, i.e. $p_\theta(x|z)$, such that the new data resembles our training data
- a.k.a generation network

- Distribution of latent variable z
  - True posterior: $p_\theta(z|x)$ not known
  - Prior: $p_\theta(z)$, initial assumption about how z is distributed

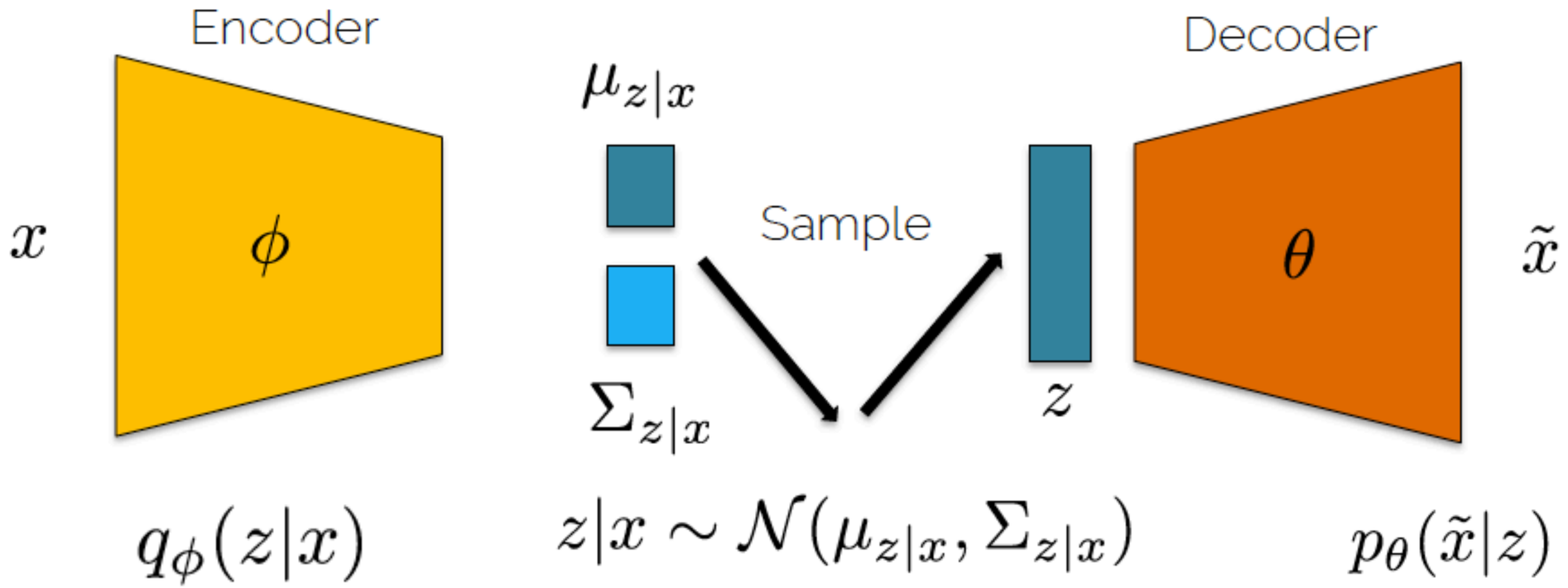
- Encoder maps input x to a **distribution** $q_\phi(z|x)$
  - In case of gaussian, the encoder outputs vectors of means and std. dev from which we sample z
- a.k.a recognition network or inference network

Variational autoencoders

Goal: Sample from the latent distribution to generate new outputs!



Variational autoencoders



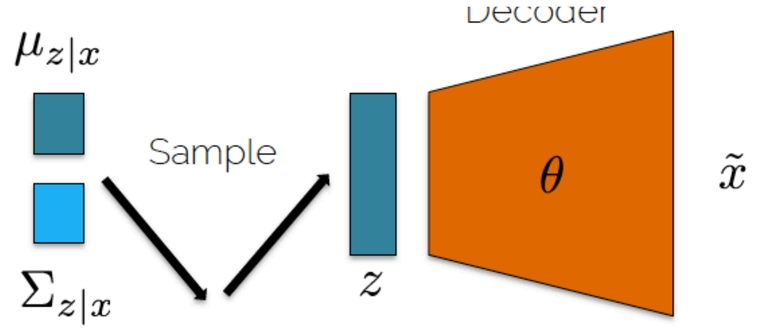
- Assume Latent space Z a Gaussian with mean \mu and covariance \Sigma
- Estimate the parameters of the generative model $p_\theta(\tilde{x}|z)$

VAE ~ MLE

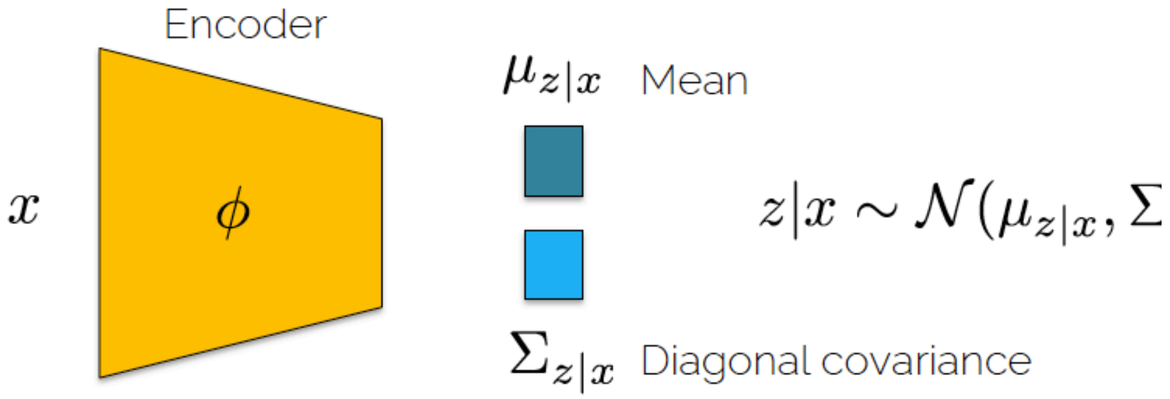
- Assume the form $\mathcal{P}_\theta$ for data distribution
- Similar to probabilistic PCA
- EM to update
  - Latent parameters (\mu, \Sigma)
  - Model parameters \theta
- Take $q_\phi(z|x)$ as the true distribution instead of $\mathcal{P}$

$$\mathop{\mathbb{E}}_{x}[\ell(\theta, x)] = -\sum_x \mathcal{P}[x] \log(\mathcal{P}_\theta[x])$$

$$= \underbrace{\sum_x \mathcal{P}[x] \log\left(\frac{\mathcal{P}[x]}{\mathcal{P}_\theta[x]}\right)}_{D_{\mathrm{RE}}[\mathcal{P}||\mathcal{P}_\theta]} + \underbrace{\sum_x \mathcal{P}[x] \log\left(\frac{1}{\mathcal{P}[x]}\right)}_{H(\mathcal{P})}$$



$$z|x \sim \mathcal{N}(\mu_{z|x}, \Sigma_{z|x})$$



$$z|x \sim \mathcal{N}(\mu_{z|x}, \Sigma$$

$$\log(p_\theta(x_i)) = \boldsymbol{E}_{z\sim q_\phi(z|x_i)}[\log(p_\theta(x_i))]$$

$$= \boldsymbol{E}_{z\sim q_\phi(z|x_i)}\left[\log \frac{p_\theta(x_i|z)p_\theta(z)}{p_\theta(z|x_i)}\right]$$

$$= \boldsymbol{E}_z\left[\log \frac{p_\theta(x_i|z)p_\theta(z)}{p_\theta(z|x_i)} \frac{q_\phi(z|x_i)}{q_\phi(z|x_i)}\right]$$

VAE ELBO maximisation

- Log-likelihood for x_i:

$$\log(p_\theta(x_i)) = \boldsymbol{E}_z \left[\log \frac{p_\theta(x_i|z)\, p_\theta(z)\, q_\phi(z|x_i)}{p_\theta(z|x_i)\, q_\phi(z|x_i)} \right]$$

$$= \boldsymbol{E}_z \left[\log p_\theta(x_i|z) \right] - \boldsymbol{E}_z \left[\log \frac{q_\phi(z|x_i)}{p_\theta(z)} \right] + \boldsymbol{E}_z \left[\log \frac{q_\phi(z|x_i)}{p_\theta(z|x_i)} \right]$$

Apply the logarithm and group as needed

Decoder | Encoder | prior

$$\underbrace{\boldsymbol{E}_z \left[\log p_\theta(x_i|z) \right] - KL(q_\phi(z|x_i)||p_\theta(z))}_{\text{Lower bound of the loss function = evidence lower bound (ELBO)}} + \underbrace{KL(q_\phi(z|x_i)||p_\theta(z|x_i))}_{\geq 0}$$

$$\mathcal{L}(x_i, \phi, \theta)$$

$$KL(q_\phi(z|x_i)||p_\theta(z)) = \int dz \quad p_\theta(z)[\log \frac{q_\phi(z|x_i)}{p_\theta(z)}] = \mathbf{E}_z[\log \frac{q_\phi(z|x_i)}{p_\theta(z)}]$$

$\int dz \quad p_\theta(z|x_i)$ is intractable but KL(...) >= 0

- Optimize $\phi^*, \theta^* = \arg\max \sum_{i=1}^{N} \mathcal{L}(x_i, \phi, \theta)$

EM as an alternating maximisation problem

$$\log \prod_{i=1}^{m} \mathcal{P}_{\boldsymbol{\theta}}[X = \mathbf{x}_i] \;\geq\; F(Q, \boldsymbol{\theta}) = \sum_{i=1}^{m} \sum_{y=1}^{k} Q_{i,y} \log \left(\mathcal{P}_{\boldsymbol{\theta}}[X = \mathbf{x}_i, Y = y]\right)$$

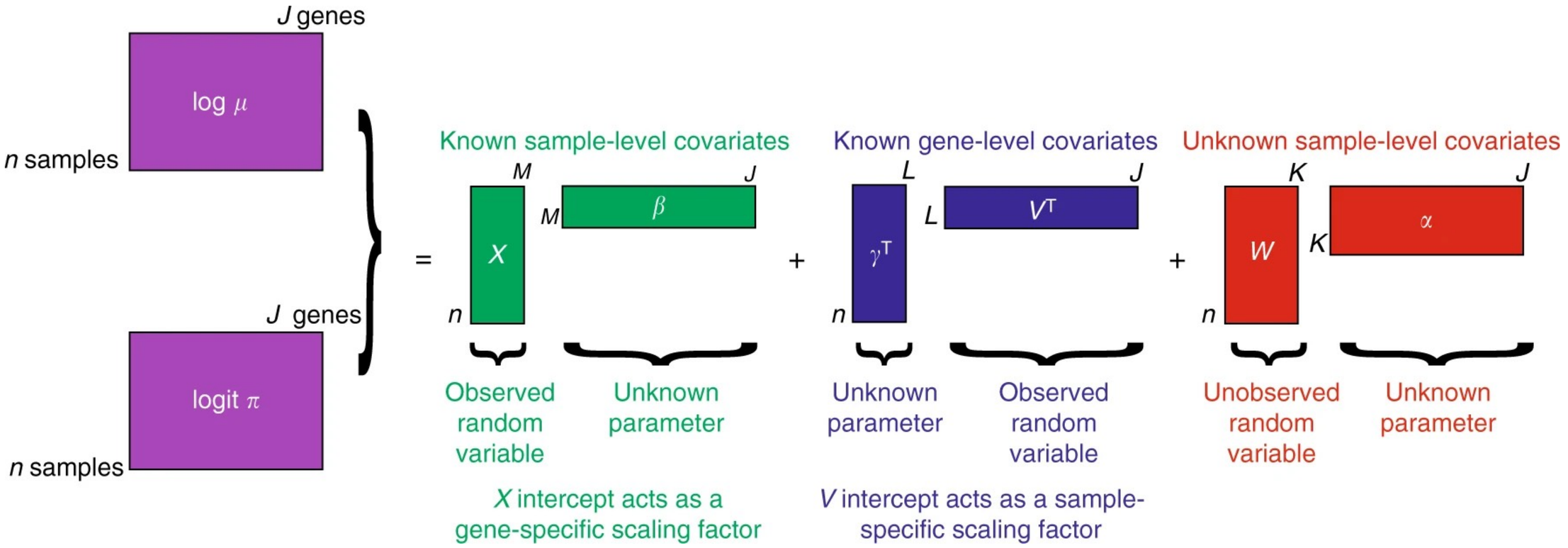
$$G(Q, \boldsymbol{\theta}) = F(Q, \boldsymbol{\theta}) - \sum_{i=1}^{m} \sum_{y=1}^{k} Q_{i,y} \log(Q_{i,y}) \qquad \mathbb{Q} = \left\{ Q \in [0,1]^{m,k} : \forall i, \; \sum_{y=1}^{k} Q_{i,y} = 1 \right\}$$

LEMMA 24.2 *The EM procedure can be rewritten as:*

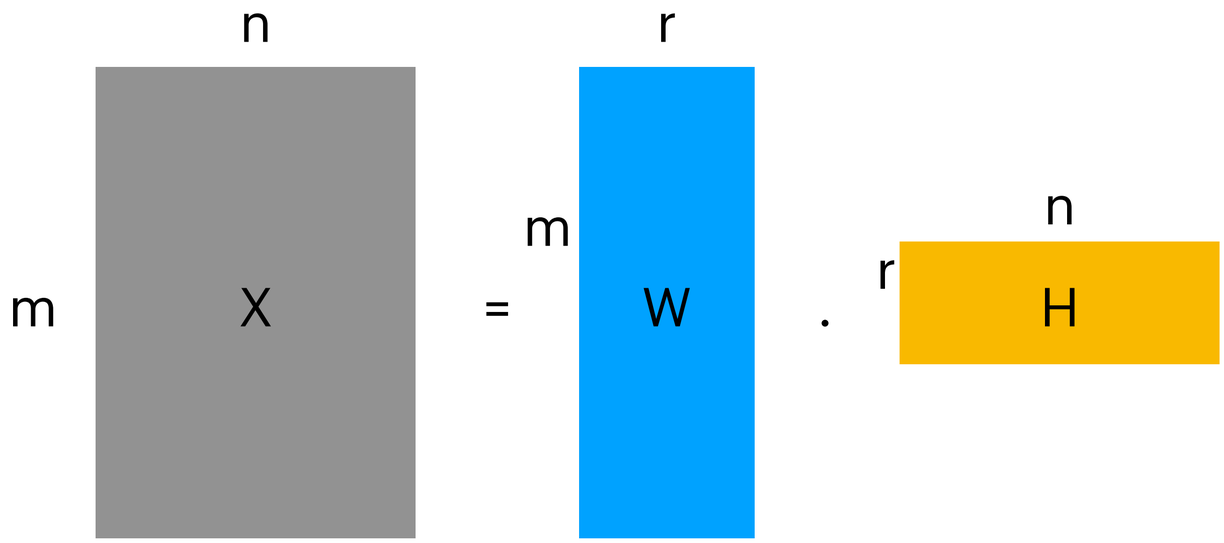
$$\begin{aligned} Q^{(t+1)} &= \operatorname*{argmax}_{Q \in \mathbb{Q}} G(Q, \boldsymbol{\theta}^{(t)}) \\ \boldsymbol{\theta}^{(t+1)} &= \operatorname*{argmax}_{\boldsymbol{\theta}} G(Q^{(t+1)}, \boldsymbol{\theta}) \; . \end{aligned}$$

Furthermore, $G(Q^{(t+1)}, \boldsymbol{\theta}^{(t)}) = L(\boldsymbol{\theta}^{(t)})$.

ZINB-WaVE (variational) matrix factorisation into known and unknown components



Matrix factorization



- Regularisation via imposing priors
- Regularisation via constraints

Non-negative Matrix Factorisation (NMF)

The exact NMF Problem : given $\mathbf{X} \in \mathbb{R}_+^{m \times n}$, find $\mathbf{W} \in \mathbb{R}_+^{m \times r}$ and $H \in \mathbb{R}_+^{r \times n}$ such that $\mathbf{X} = \mathbf{WH}$.

- A very difficult problem : NP-hard !

The approximate NMF problem: given $\mathbf{X} \in \mathbb{R}_+^{m \times n}$, find $\mathbf{W} \in \mathbb{R}_+^{m \times r}$ and $H \in \mathbb{R}_+^{r \times n}$ such that $\mathbf{X} \approx \mathbf{WH}$.

i.e., given $\mathbf{X} \in \mathbb{R}_+^{m \times n}$, find $\mathbf{W}$ and $\mathbf{H}$ that minimizes

$$\|\mathbf{X} - \mathbf{WH}\|_\xi$$

where $\|.\|_\xi$ is some norm that measures the discrepancy between $\mathbf{X}$ and $\mathbf{WH}$.

There are many types of $\|.\|_\xi$, here we consider Frobenius norm, that is

$$\|\mathbf{X} - \mathbf{WH}\|_F$$

The NMF constrained optimisation problem

$$[\mathbf{W}\ \mathbf{H}] = \underset{\mathbf{W}\geq 0, \mathbf{H}\geq 0}{\arg\min}\ f\left(\mathbf{W}, \mathbf{H}\right) = \frac{1}{2}\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2$$

We solve the problem using alternating Gradient Descent (GD).

We first review GD. GD is a first-order iterative optimization algorithm.

For minimizing a single variable function $f(x)$, GD starts with an inital x_0 and iterates the following update:

$$x^{k+1} = x^k - t^k \nabla_x f(x^k)$$

where

- $k \in \mathbb{N}$ is the step counter
- x^k is the current variable
- x^{k+1} is the variable of the next iteration
- $t^k \geq 0$ is the step size
- $\nabla_x f(x)$ is the gradient of the objective function f wit respect to x

Optimization via alternating minimisation or block coordinate descent

Method: alternating GD

$$\mathbf{W}^{k+1} = \mathbf{W}^k - t^k_{\mathbf{W}} \nabla_{\mathbf{W}} f(\mathbf{W}^k, \mathbf{H}^k)$$

$$\mathbf{H}^{k+1} = \mathbf{H}^k - t^k_{\mathbf{H}} \nabla_{\mathbf{H}} f(\mathbf{W}^{k+1}, \mathbf{H}^k)$$

To apply alternating GD, we need to know

- $\nabla_{\mathbf{W}} f$ and $\nabla_{\mathbf{H}} f$
- $t^k_{\mathbf{W}}$, $t^k_{\mathbf{H}}$

$\nabla_{\mathbf{W}} f$ and $\nabla_{\mathbf{H}} f$

f= $\|\mathbf{X} - \mathbf{WH}\|_F^2 \overset{(1)}{=} \operatorname{tr}\left\{\left(\mathbf{X} - \mathbf{WH}\right)^{\intercal}\left(\mathbf{X} - \mathbf{WH}\right)\right\}$

$= \operatorname{tr}\left\{\mathbf{X}^\top\mathbf{X} - \mathbf{X}^\top\mathbf{WH} - (\mathbf{WH})^\top\mathbf{X} + (\mathbf{WH})^\top\mathbf{WH}\right\}$

$\overset{(2)}{=} \operatorname{tr}\left(\mathbf{X}^\top\mathbf{X} - 2(\mathbf{WH})^\top\mathbf{X} + \mathbf{H}^\top\mathbf{W}^\top\mathbf{WH}\right)$

$\overset{(1)}{=} \|\mathbf{X}\|_F^2 - 2\operatorname{tr}(\mathbf{WH})^\top\mathbf{X} + \operatorname{tr}\mathbf{H}^\top\mathbf{W}^\top\mathbf{WH}$

Multiplicative gradient descent

So $f(\mathbf{W},\mathbf{H}) = \frac{1}{2}\|\mathbf{X}-\mathbf{WH}\|_F^2 = \|\mathbf{X}\|_F^2 - \operatorname{tr}\mathbf{X}^\top\mathbf{WH} + \frac{1}{2}\operatorname{tr}\mathbf{H}^\top\mathbf{W}^\top\mathbf{WH}$,
and

$$\begin{aligned}\nabla_{\mathbf{W}} f &\overset{(3,5)}{=} (\mathbf{WH}-\mathbf{X})\mathbf{H}^\top \\ \nabla_{\mathbf{H}} f &\overset{(3,4)}{=} \mathbf{W}^\top(\mathbf{WH}-\mathbf{X})\end{aligned}$$

So we get

$$\begin{aligned}\mathbf{W}^{k+1} &= \mathbf{W}^k - t_{\mathbf{W}}^k(\mathbf{WH}-\mathbf{X})\mathbf{H}^\top \\ \mathbf{H}^{k+1} &= \mathbf{H}^k - t_{\mathbf{H}}^k\mathbf{W}^\top(\mathbf{WH}-\mathbf{X})\end{aligned}$$

- Choose t^k (learning rate) such that the update rule becomes multiplicative
- Different learning rates for each matrix element (ij)

$$[t_{\mathbf{W}}]_{ij} = \frac{[\mathbf{W}]_{ij}}{[\mathbf{WHH}^\top]_{ij}} \qquad [t_{\mathbf{H}}]_{ij} = \frac{[\mathbf{H}]_{ij}}{[\mathbf{W}^\top\mathbf{WH}]_{ij}}$$

Imposing the non-negativity constraint

- Non-negative initialisation —> non-negative final solution

$$
\begin{aligned}
[\mathbf{W}]_{ij} &= [\mathbf{W}]_{ij} - [t_{\mathbf{W}}]_{ij} \left[(\mathbf{WH}-\mathbf{X})\mathbf{H}^\top\right]_{ij} \\
&= [\mathbf{W}]_{ij} - \frac{[\mathbf{W}]_{ij}}{[\mathbf{WHH}^\top]_{ij}} \left[(\mathbf{WH}-\mathbf{X})\mathbf{H}^\top\right]_{ij} \\
&= [\mathbf{W}]_{ij} - \frac{\left[\mathbf{W}(\mathbf{WH}-\mathbf{X})\mathbf{H}^\top\right]_{ij}}{[\mathbf{WHH}^\top]_{ij}} \\
&= [\mathbf{W}]_{ij} \left(1 - \frac{\left[(\mathbf{WH}-\mathbf{X})\mathbf{H}^\top\right]_{ij}}{[\mathbf{WHH}^\top]_{ij}}\right) \\
&= [\mathbf{W}]_{ij} \left(\frac{[\mathbf{WHH}^\top]_{ij}}{[\mathbf{WHH}^\top]_{ij}} - \frac{\left[(\mathbf{WH}-\mathbf{X})\mathbf{H}^\top\right]_{ij}}{[\mathbf{WHH}^\top]_{ij}}\right) \\
&= [\mathbf{W}]_{ij} \frac{\left[\mathbf{XH}^\top\right]_{ij}}{[\mathbf{WHH}^\top]_{ij}}
\end{aligned}
$$

$$
\begin{aligned}
[\mathbf{H}]_{ij} &= [\mathbf{H}]_{ij} - [t_{\mathbf{H}}]_{ij} \left[\mathbf{W}^\top(\mathbf{WH}-\mathbf{X})\right]_{ij} \\
&= [\mathbf{H}]_{ij} - \frac{[\mathbf{H}]_{ij}}{[\mathbf{W}^\top\mathbf{WH}]_{ij}} \mathbf{W}^\top(\mathbf{WH}-\mathbf{X}) \\
&= [\mathbf{H}]_{ij} - \frac{[\mathbf{H}]_{ij}}{[\mathbf{W}^\top\mathbf{WH}]_{ij}} \left[\mathbf{W}^\top(\mathbf{WH}-\mathbf{X})\right]_{ij} \\
&= [\mathbf{H}]_{ij} \left(1 - \frac{\left[\mathbf{W}^\top(\mathbf{WH}-\mathbf{X})\right]_{ij}}{[\mathbf{W}^\top\mathbf{WH}]_{ij}}\right) \\
&= [\mathbf{H}]_{ij} \left(\frac{[\mathbf{W}^\top\mathbf{WH}]_{ij}}{[\mathbf{W}^\top\mathbf{WH}]_{ij}} - \frac{\left[\mathbf{W}^\top(\mathbf{WH}-\mathbf{X})\right]_{ij}}{[\mathbf{W}^\top\mathbf{WH}]_{ij}}\right) \\
&= [\mathbf{H}]_{ij} \frac{\left[\mathbf{W}^\top\mathbf{X}\right]_{ij}}{[\mathbf{W}^\top\mathbf{WH}]_{ij}}
\end{aligned}
$$

$$
\mathbf{W}^{k+1} = \mathbf{W}^k \circ \frac{\mathbf{XH}^\top}{\mathbf{WHH}^\top} \qquad \mathbf{H}^{k+1} = \mathbf{H}^k \circ \frac{{\mathbf{W}^k}^\top\mathbf{X}}{(\mathbf{W}^k)^\top\mathbf{W}^k\mathbf{H}^k}
$$

Extracting multi-modal information from scRNA-seq

- Big data concept
- Sound of the keyboard —> text string, room temperature, etc.
- Correspondence between feature sets

- More efficiently using the available data
- Cost efficient experiments
- Study of rare samples

- scRNA-seq as the most available data modality
- Noisy, uncertain scRNA-seq SNV for clonal tracing?
  - Germline variants
  - Allele frequency drop-out
  - Cell type specific expression
  - Sequencing and alignment errors
  - RNA-edits

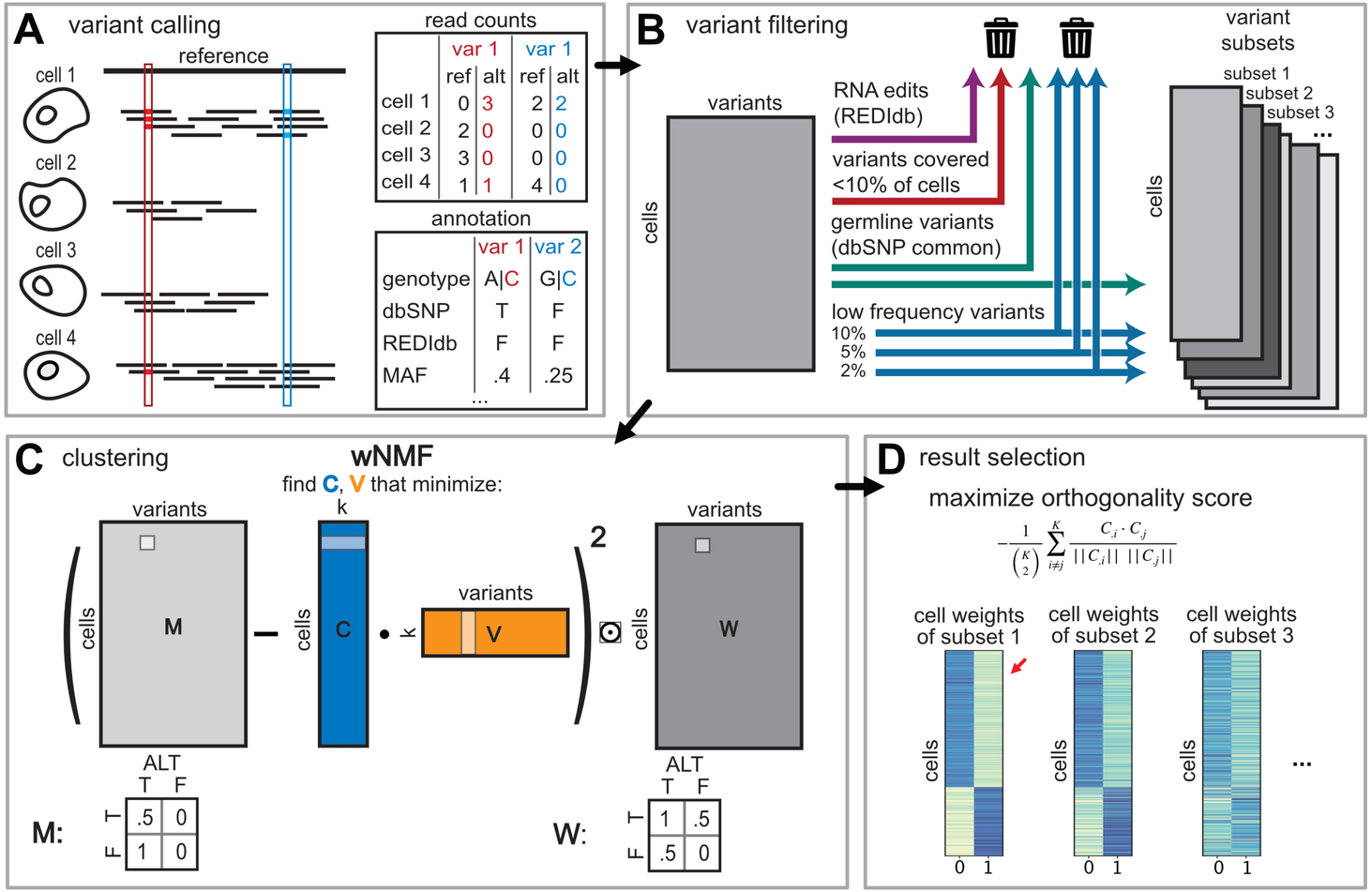
CCLONE: Clonal tracing from scRNA-seq variant calls

Gene expression

Identifying cancer cells from calling single-nucleotide variants in scRNA-seq data

Valérie Marot-Lassauzaie [1,2], Sergi Beneyto-Calabuig[3,4], Benedikt Obermayer[5], Lars Velten[3,4], Dieter Beule[5,6], Laleh Haghverdi [1,*]

- NMF for noisy data
- Weighted NMF for allelic or **cell type dependent** dropout and heteroplasty (e.g. Wij=0 if position j not expresses in cell i)
- SNVs from (high coverage) scRNA-seq data can be used for clonal inference



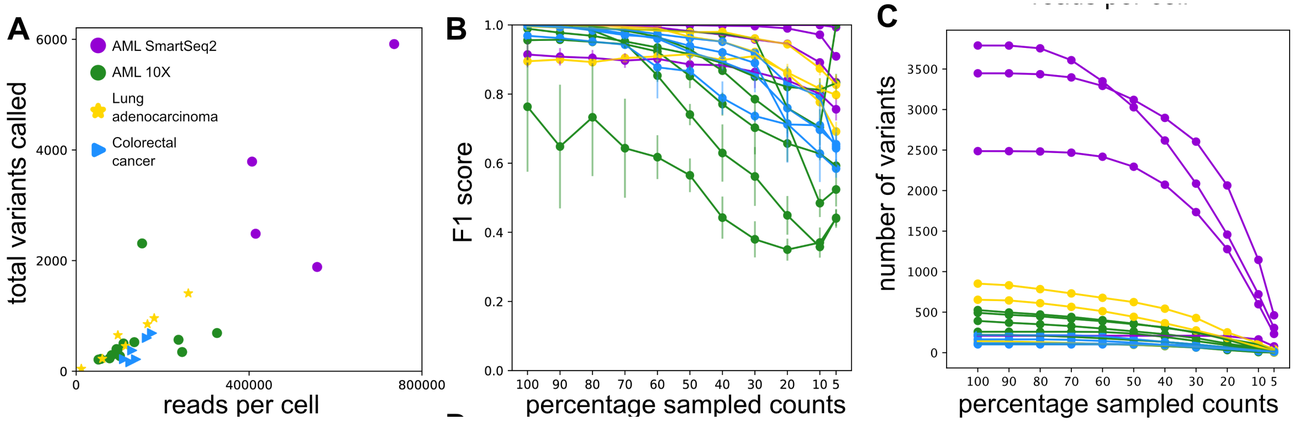
- Identifying somatic mutations in scRNA-seq and scATAC-seq:
  - Scomatic Muyas et al. Nat. Biotechnol. 2024 (comparing variant distribution in tumour and healthy cells)
  - Monopogene Dou et al. Nat. Biotechnol. 2024 (using linkage disequilibrium for distinguishing germline variants)

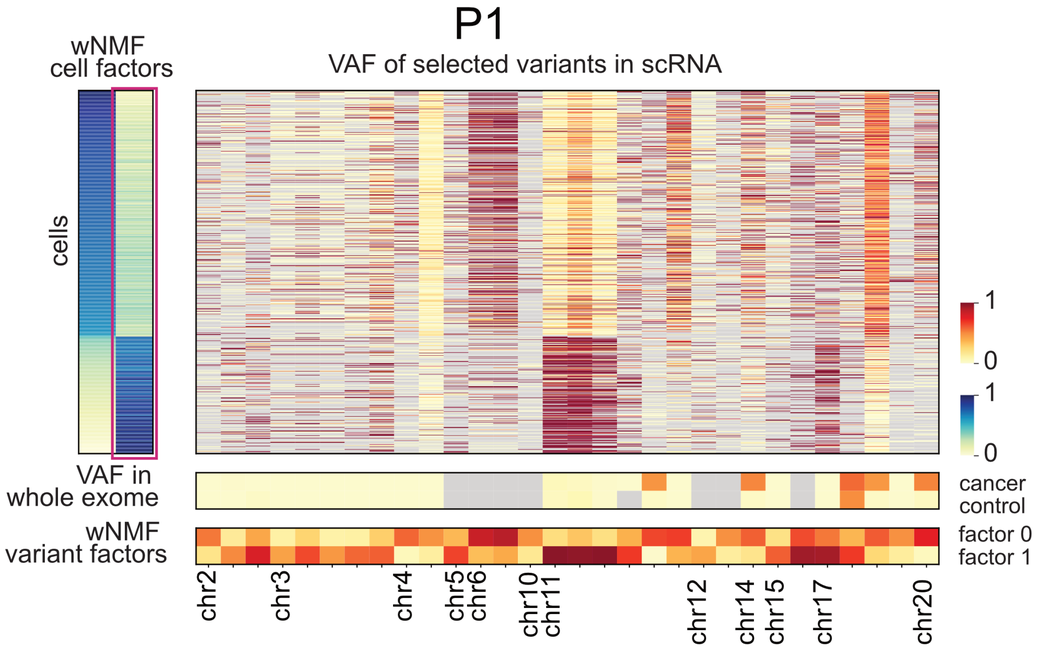
- M: Variant Allele Frequency
- W: Certainty of observation

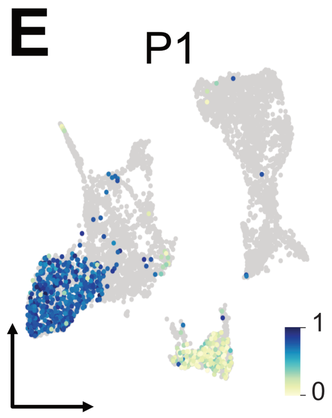
- C.i : ith column of C
- K: # of factors

CCLONE: Cancer Cell Labelling On Noisy Expression

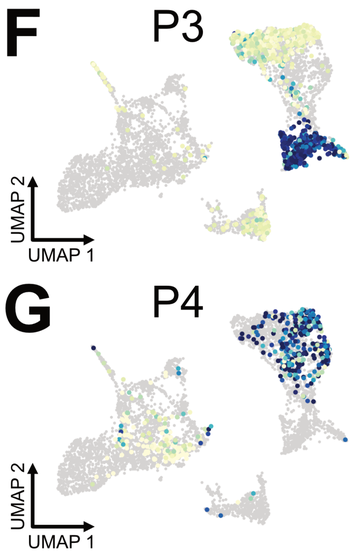
- Used for comparison as ground-truth: targeted sc-sequencing
- Higher success rate for higher mutation burden and sequencing coverage
- Interpretable factors and associated variants
- Interestingly some SNVs were not detected in WE (due to small population size, capture efficiency, different allelic expression in cancer vs. health…)

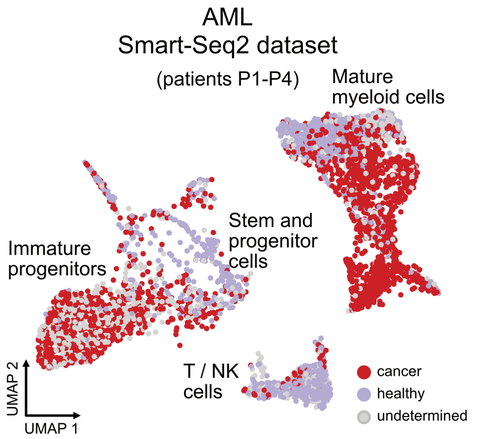






Cancer associated factor





Thank you for your attention!