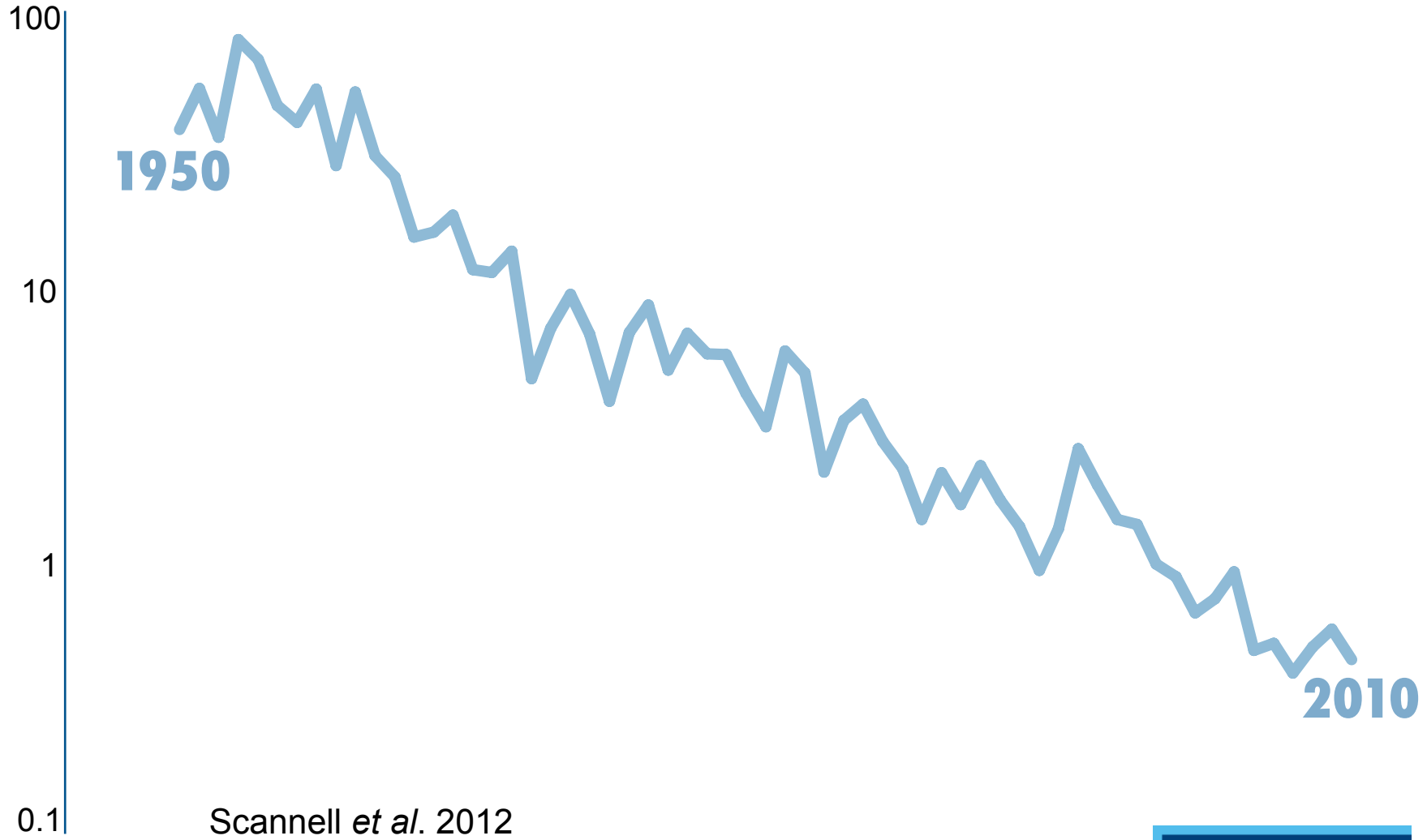
Bayesian matrix factorization for drug-target activity prediction

Yves Moreau

University of Leuven – ESAT-STADIUS
SymBioSys Center for Computational Biology

KU LEUVEN

Number of new drugs per billion US$

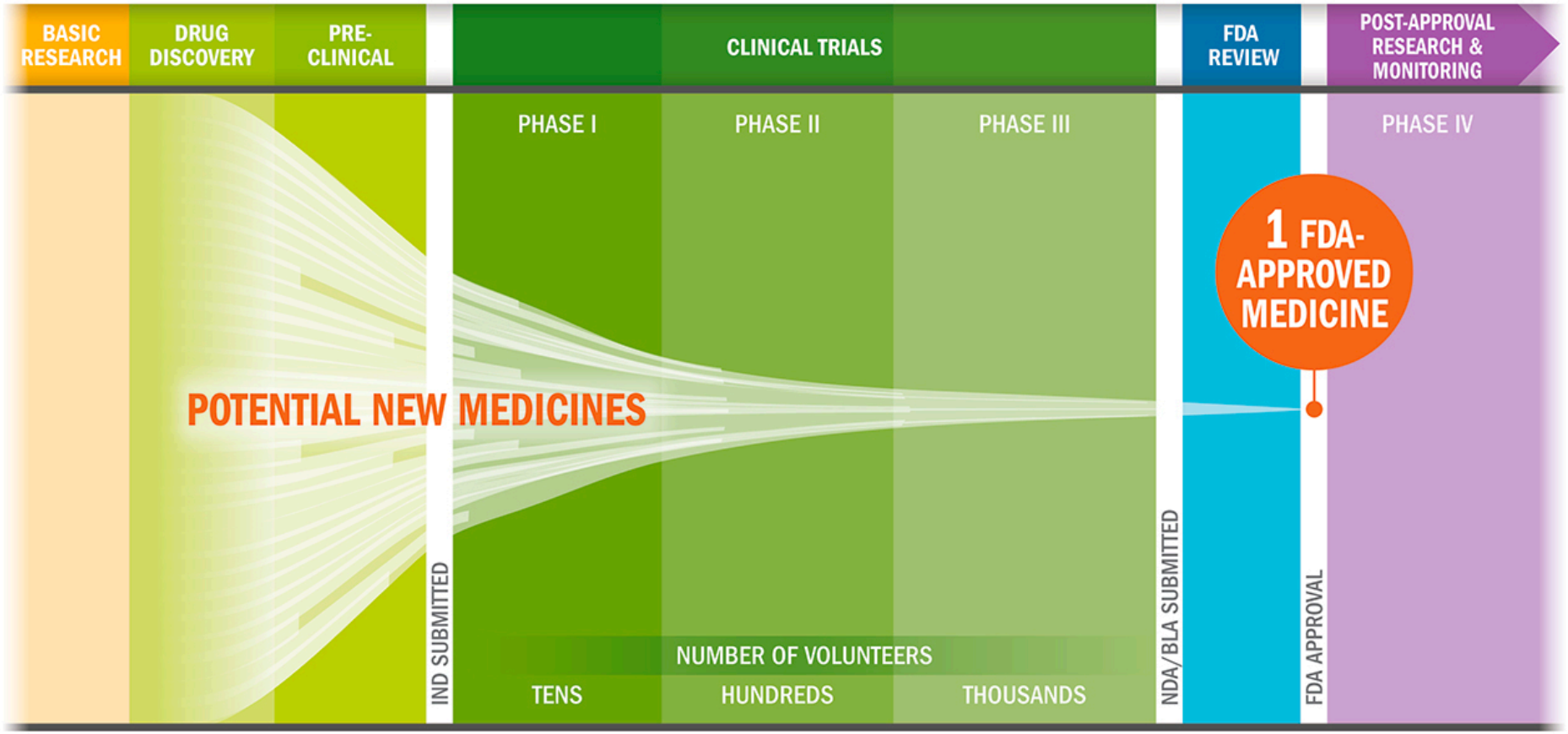


Scannell *et al*. 2012

KU LEUVEN

THE BIOPHARMACEUTICAL RESEARCH AND DEVELOPMENT PROCESS

From drug discovery through FDA approval, developing a new medicine takes at least 10 years on average and costs an average of $2.6 billion.* Less than 12% of the candidate medicines that make it into Phase I clinical trials will be approved by the FDA.

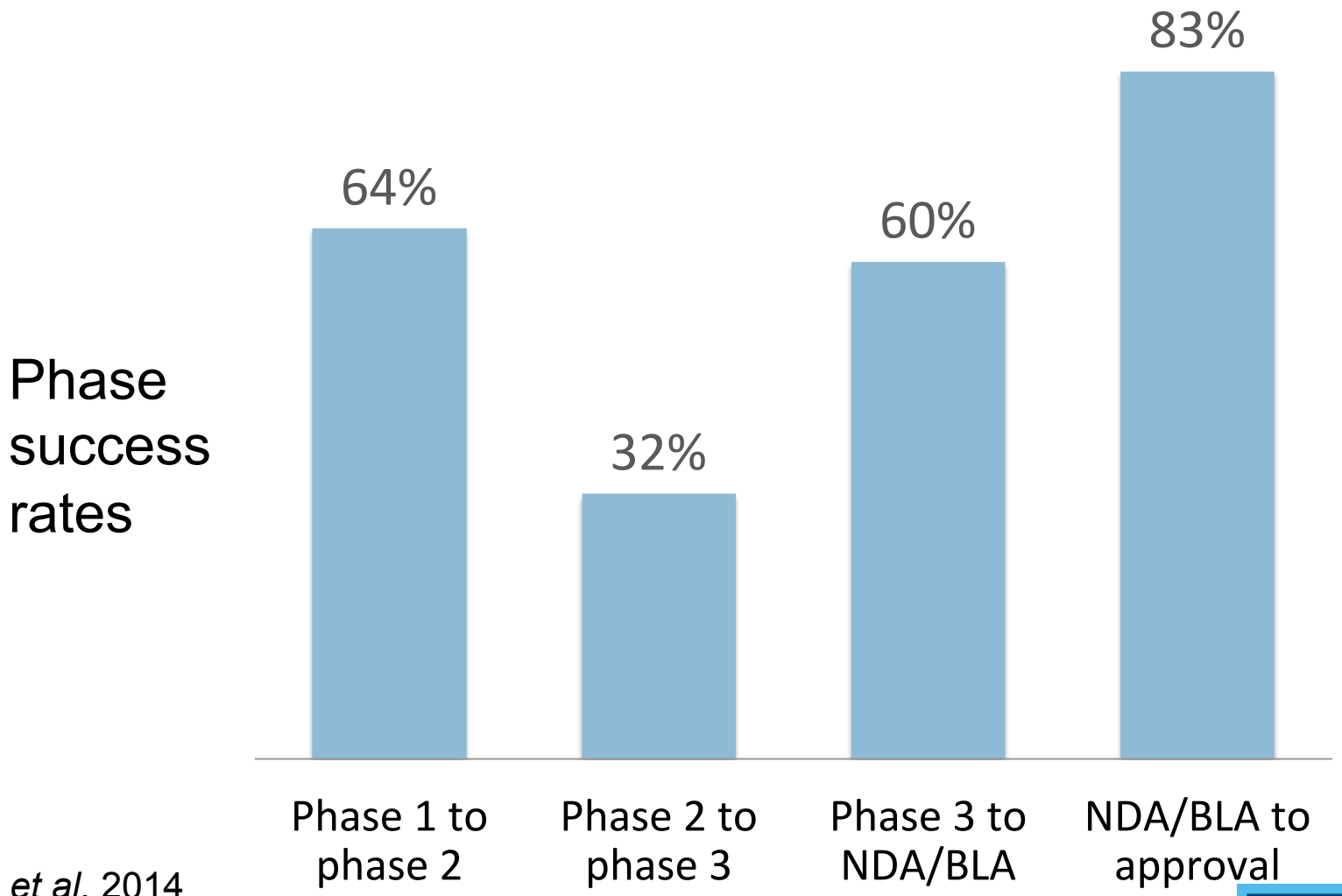


Key: IND: Investigational New Drug Application, NDA: New Drug Application, BLA: Biologics License Application

* The average R&D cost required to bring a new, FDA-approved medicine to patients is estimated to be $2.6 billion over the past decade (in 2013 dollars), including the cost of the many potential medicines that do not make it through to FDA approval.

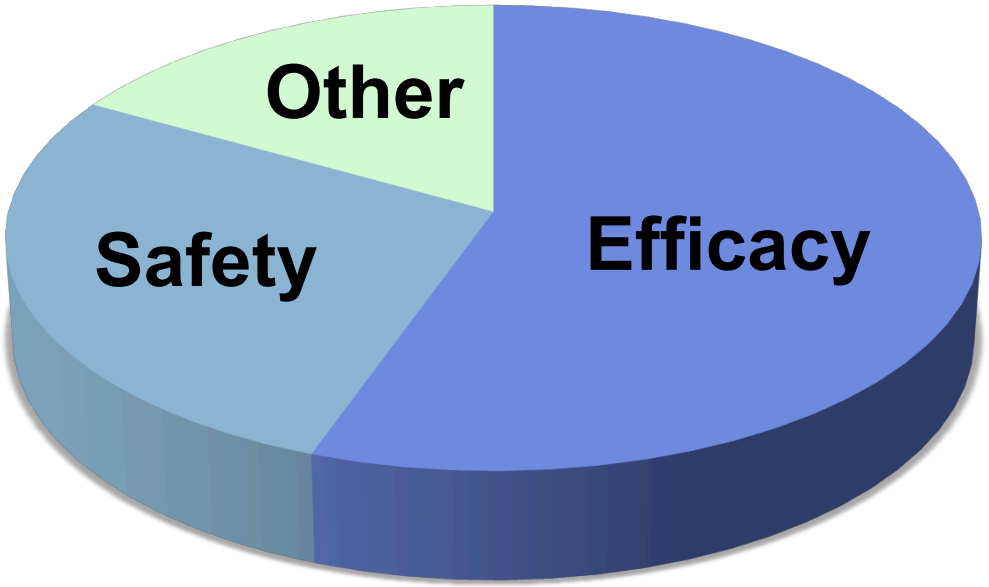
Source: PhRMA adaptation based on Tufts Center for the Study of Drug Development (CSDD) Briefing: "Cost of Developing a New Drug," Nov. 2014. Tufts CSDD & School of Medicine., and US FDA Infographic, "Drug Approval Process," http://www.fda.gov/downloads/Drugs/ResourcesForYou/Consumers/UCM284393.pdf (accessed Jan. 20, 2015).

The curse of attrition…



Hay *et al.* 2014

…mainly due to safety and efficacy issues

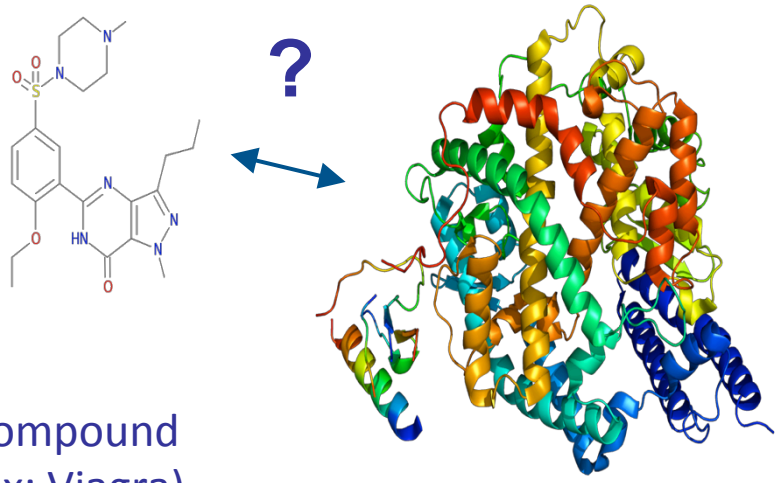


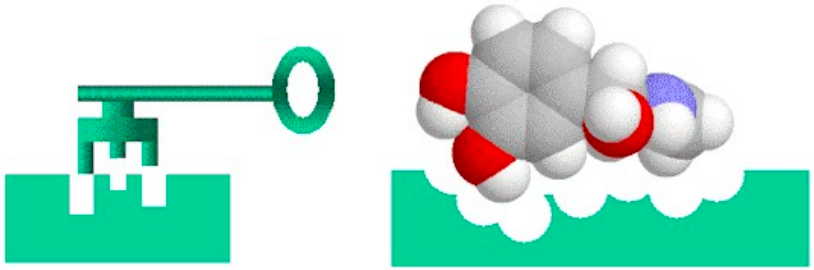
Causes of failure between Phase 2 and submission in 2011 and 2012

Arrowsmith & Miller 2013

Chemoinformatics

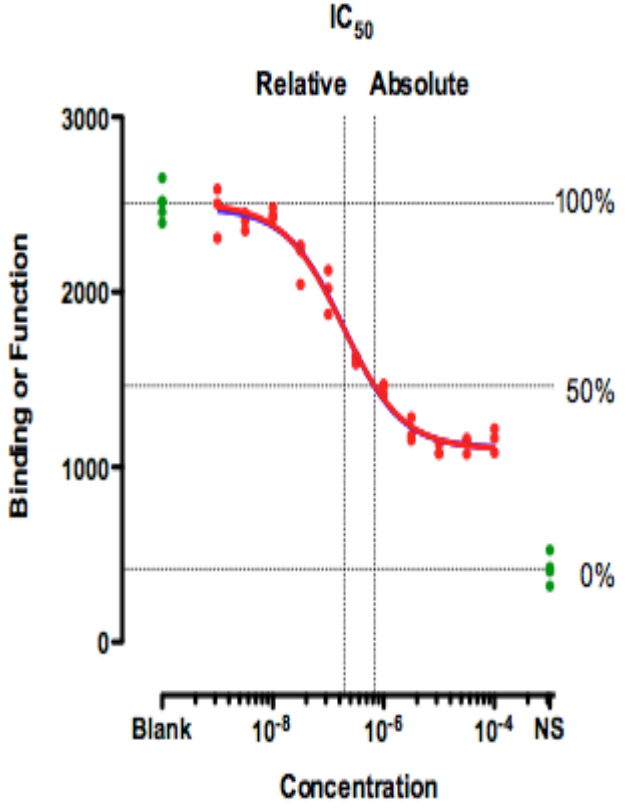
- Goal: estimate interaction between **compounds** and **protein targets**
- Activity measured by high-throughput screening
- Activity depends on match between shape of compound and shape of protein
- 3D modeling is challenging





Drug–target activities

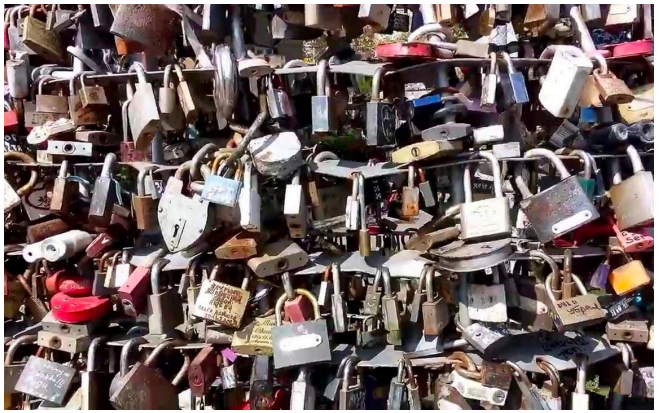
- IC50 – amount of compound needed for half **inhibition**
  - pIC50 = -log10(IC50)
- EC50 – amount of compound needed for half **effect**

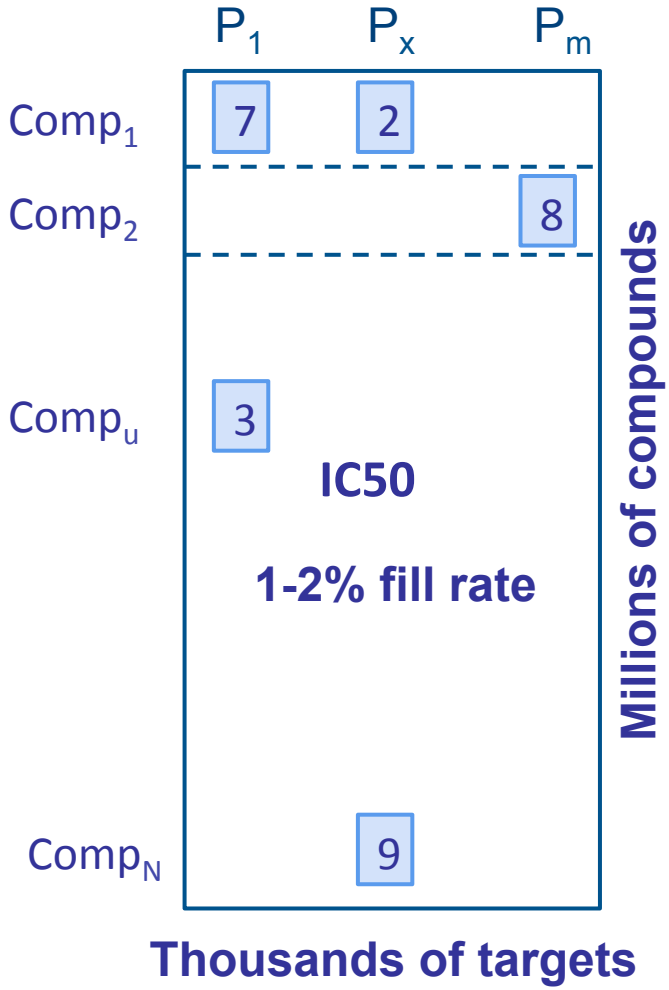


KU LEUVEN

High-throughput screening

- Hit discovery in early drug discovery
  - Identify compounds active against a protein drug target of interest
- Activity measured by high-throughput screening
- Activity = “scarce” data



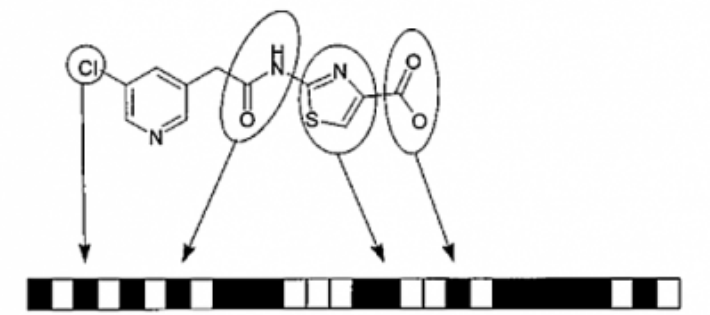


Molecular fingerprints

- High-dimensional fingerprints of 2D compound structures
- Sparse vectors

Key-based fingerprints

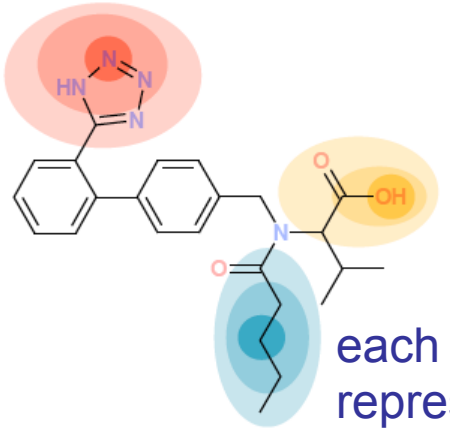
FP2 & MACCS



A bit string represents the presence or absence of particular substructures

Circular fingerprints

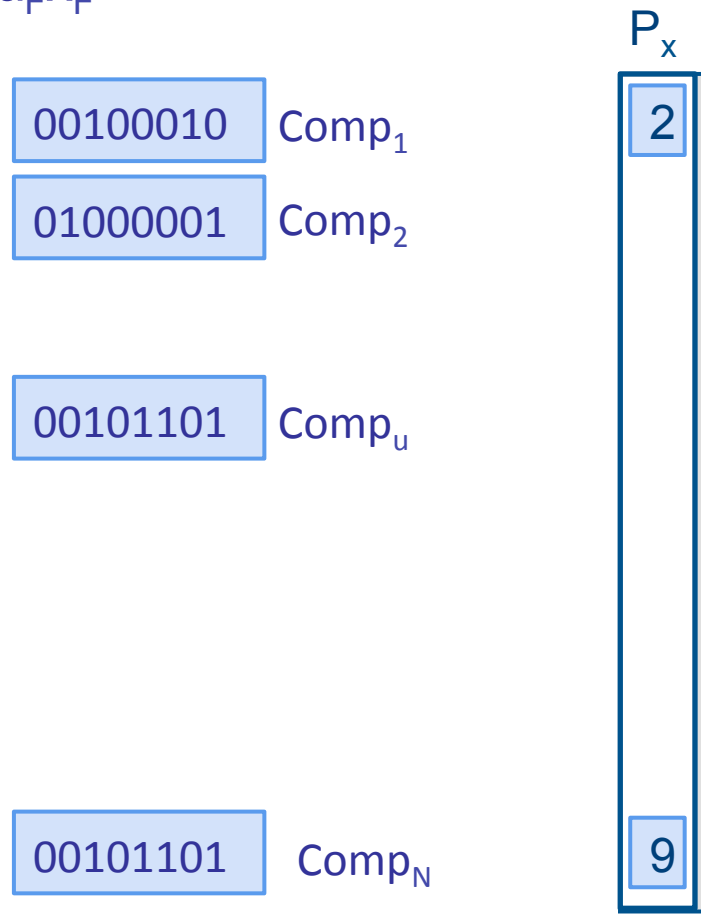
MNA & MPD & ECFP



each fingerprint represents a central atom and its neighbors

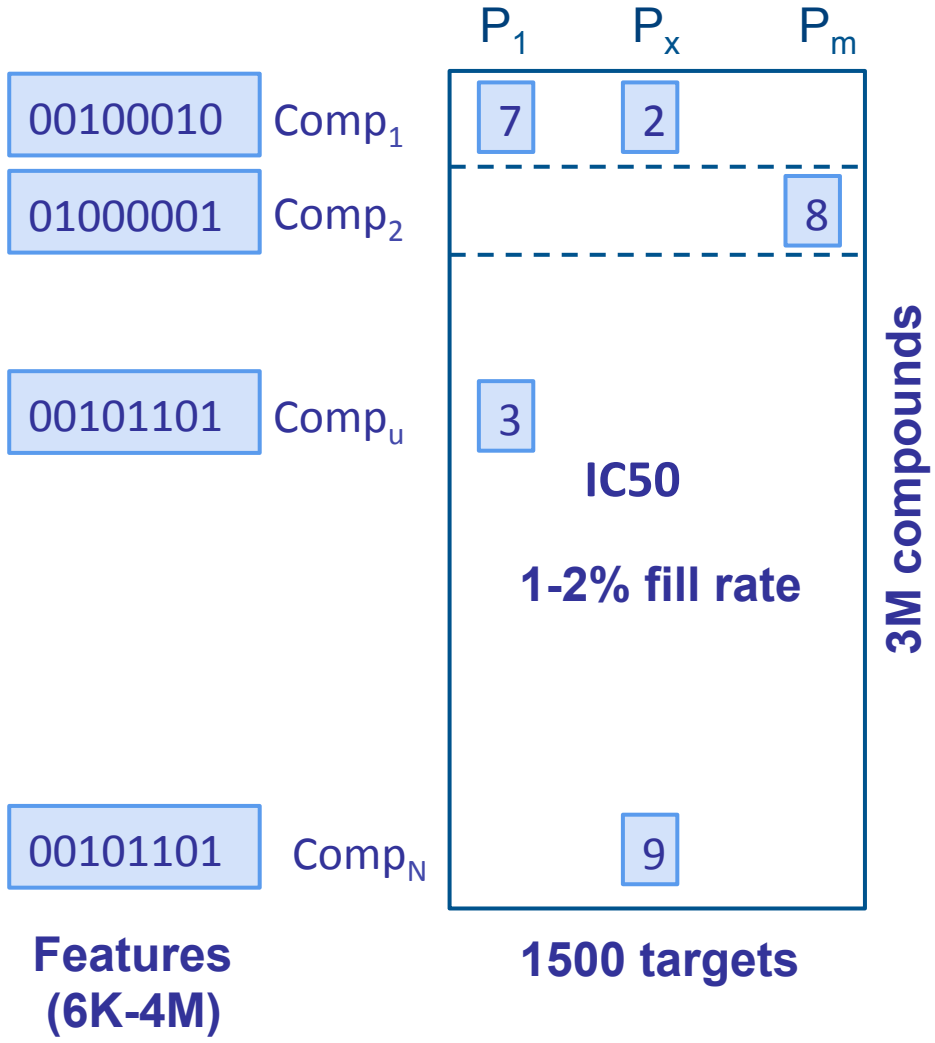
Quantitative Structure–Activity Relationship (QSAR)

- Finds optimal model **α** based on predictive features
  - $IC50(\mathbf{x}) = \alpha_1 x_1 + \alpha_2 x_2 + \ldots + \alpha_F x_F$
  - Minimize error loss
  - PLS, ridge regression
- Good performance if enough training examples
- *Does not share information across tasks!*



Multitask learning

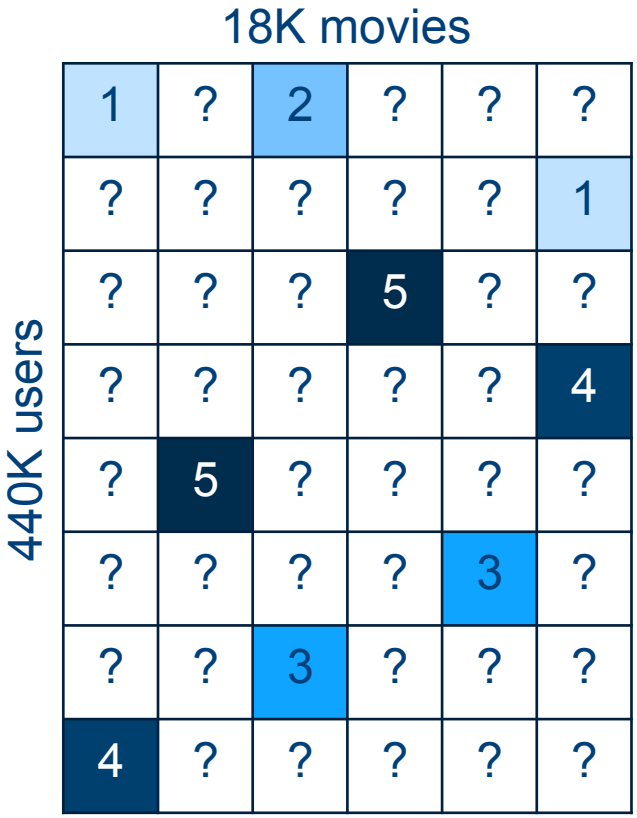
- From fingerprints and available activities, predict missing activities
- Approaches
  1. Supervised learning per target (QSAR)
  2. Matrix factorization
     - *Netflix style*
  3. MF + supervised
     - *Macau*



The Netflix Challenge

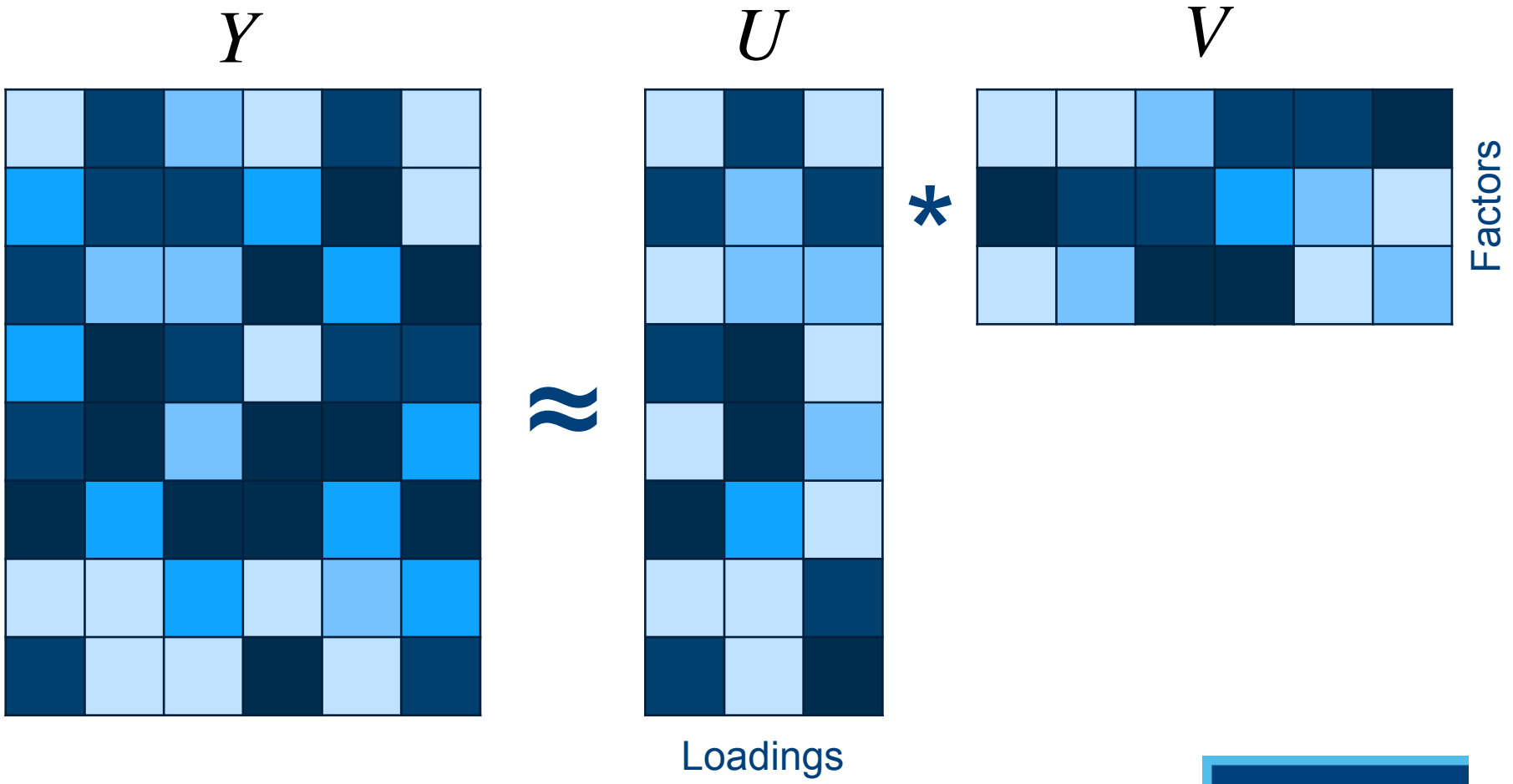
- Goal: predict user movie ratings
  - 440K users, 18K movies
  - 100 million ratings
  - 1% fill rate
  - ➔ Predict 99% missing

- *How can this work?*

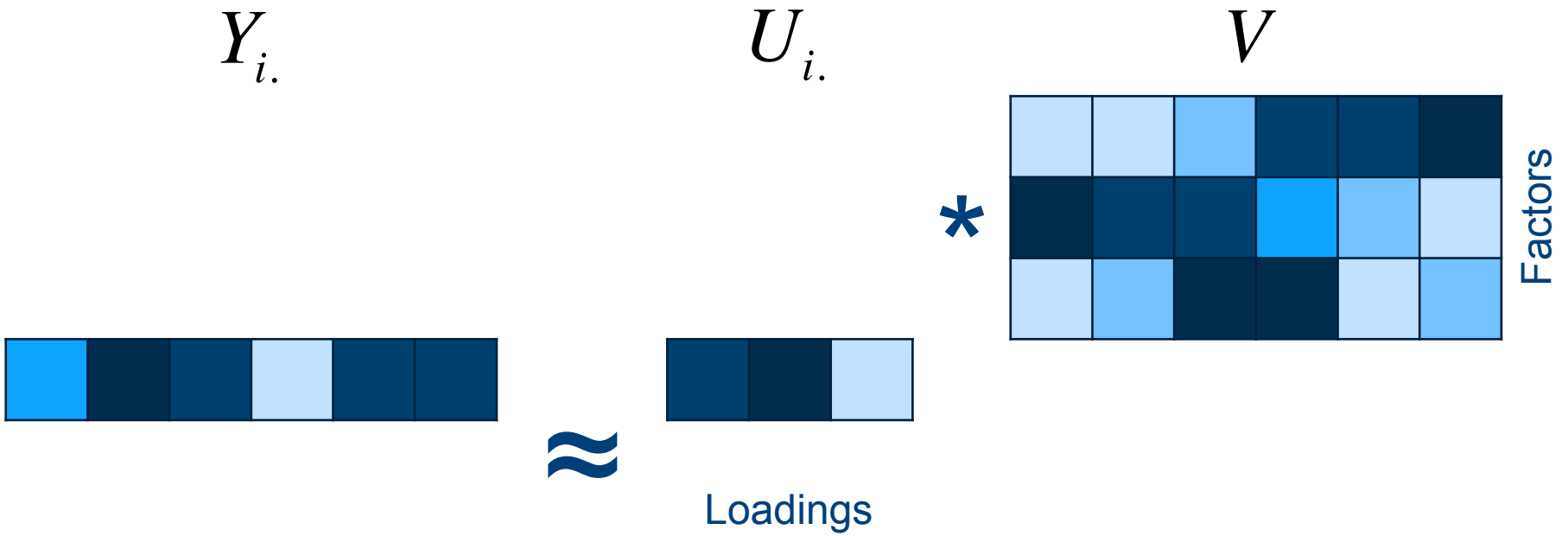


Factor analysis

- Low-rank approximation of full matrix

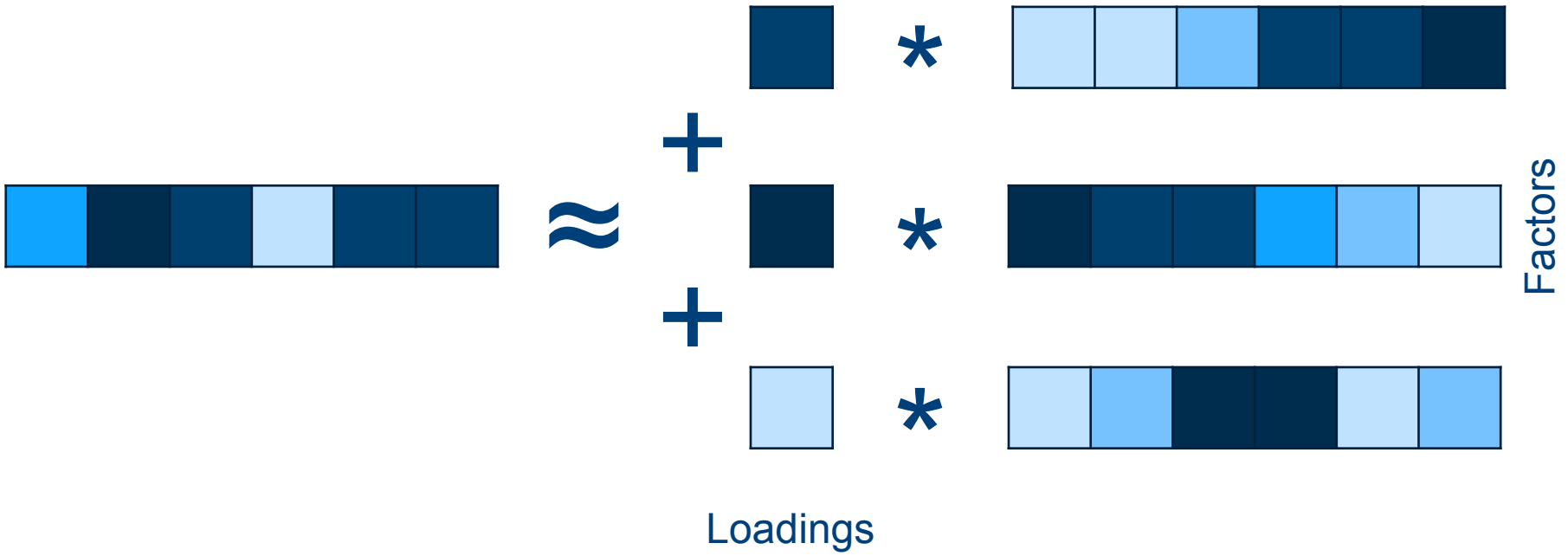


Factor analysis

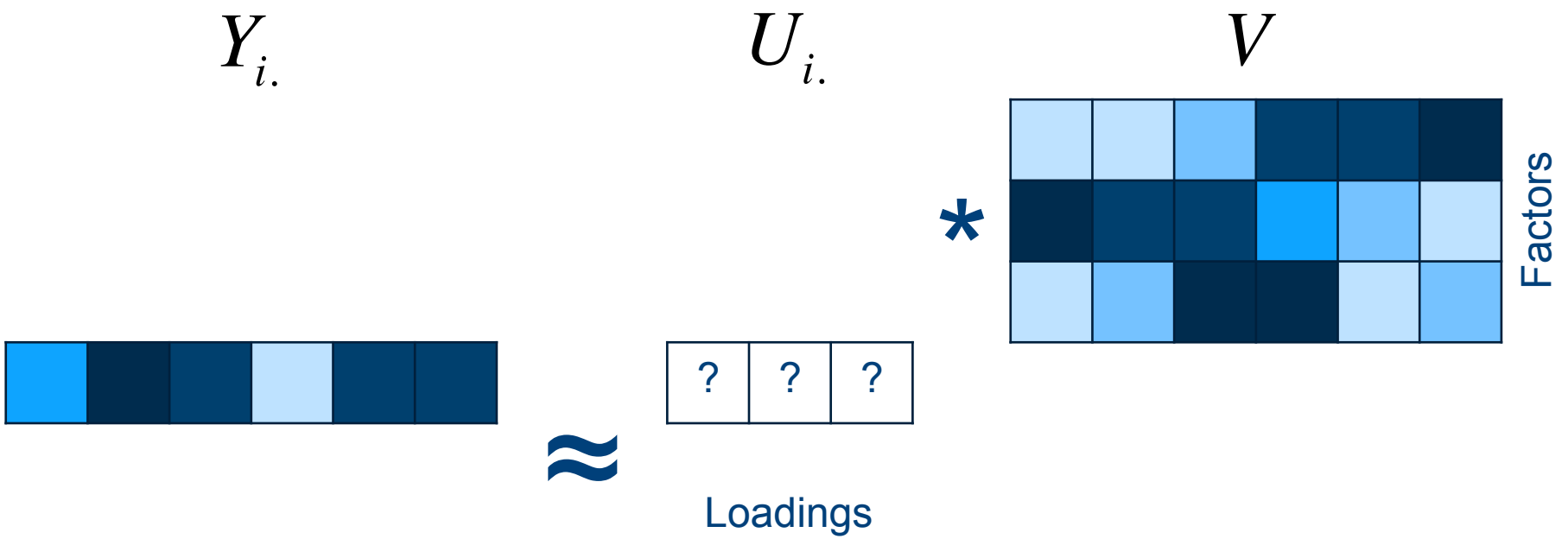


Factor analysis

- Individual response (= row) modeled as individual mixture (= loading) of a small number of latent responses (= factor)

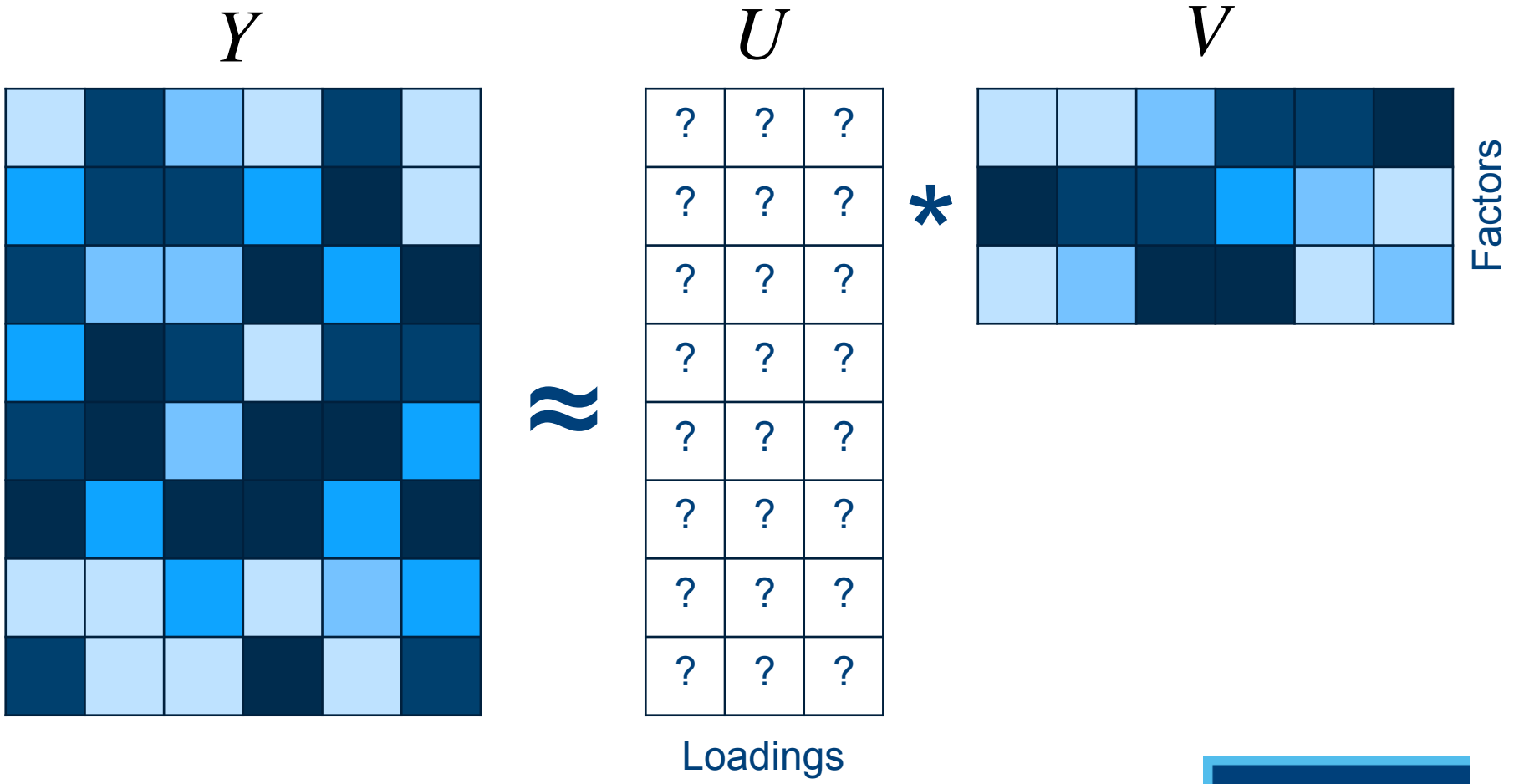


Alternating Least Squares



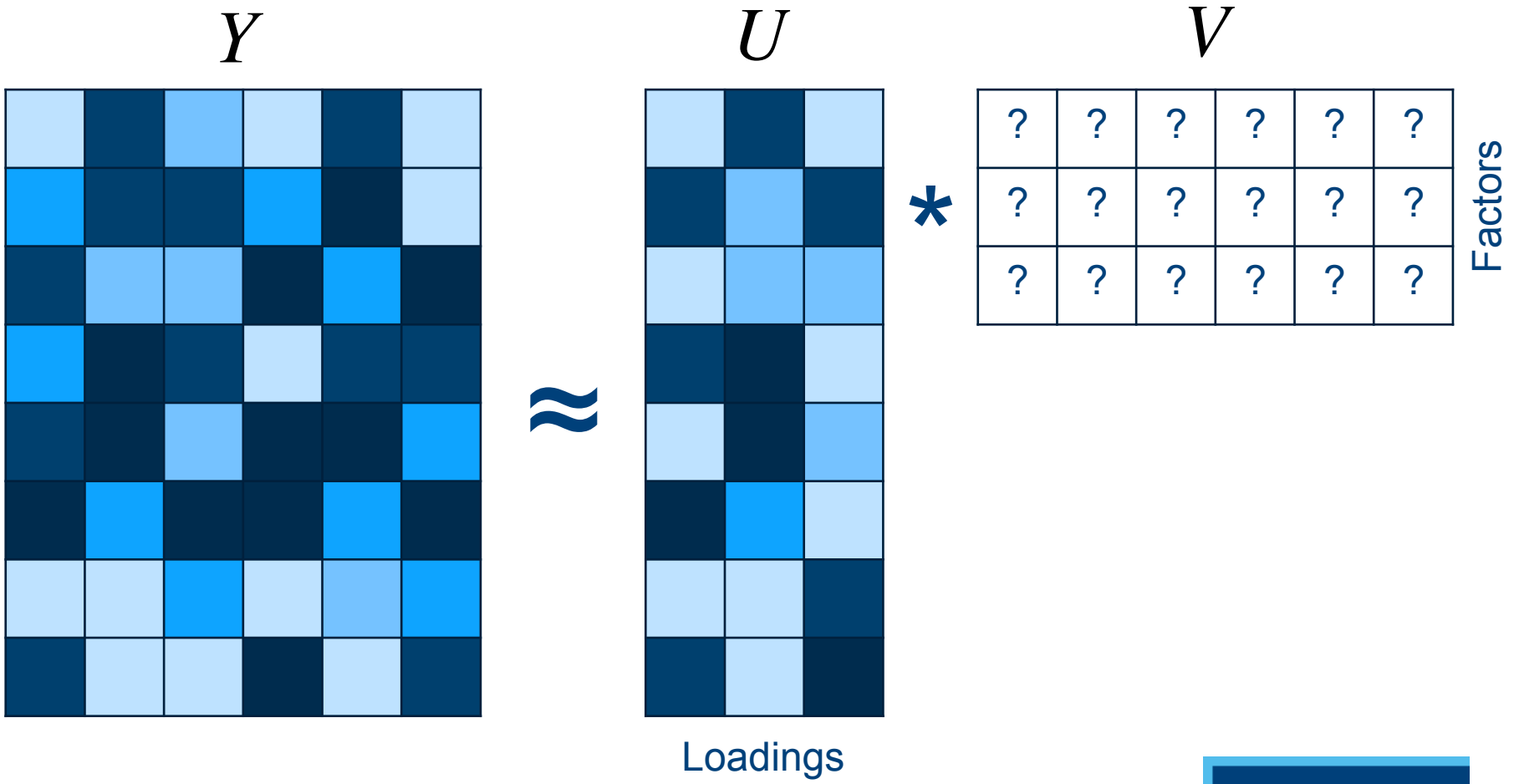
Alternating Least Squares

- If V were known, U could be found by linear regression



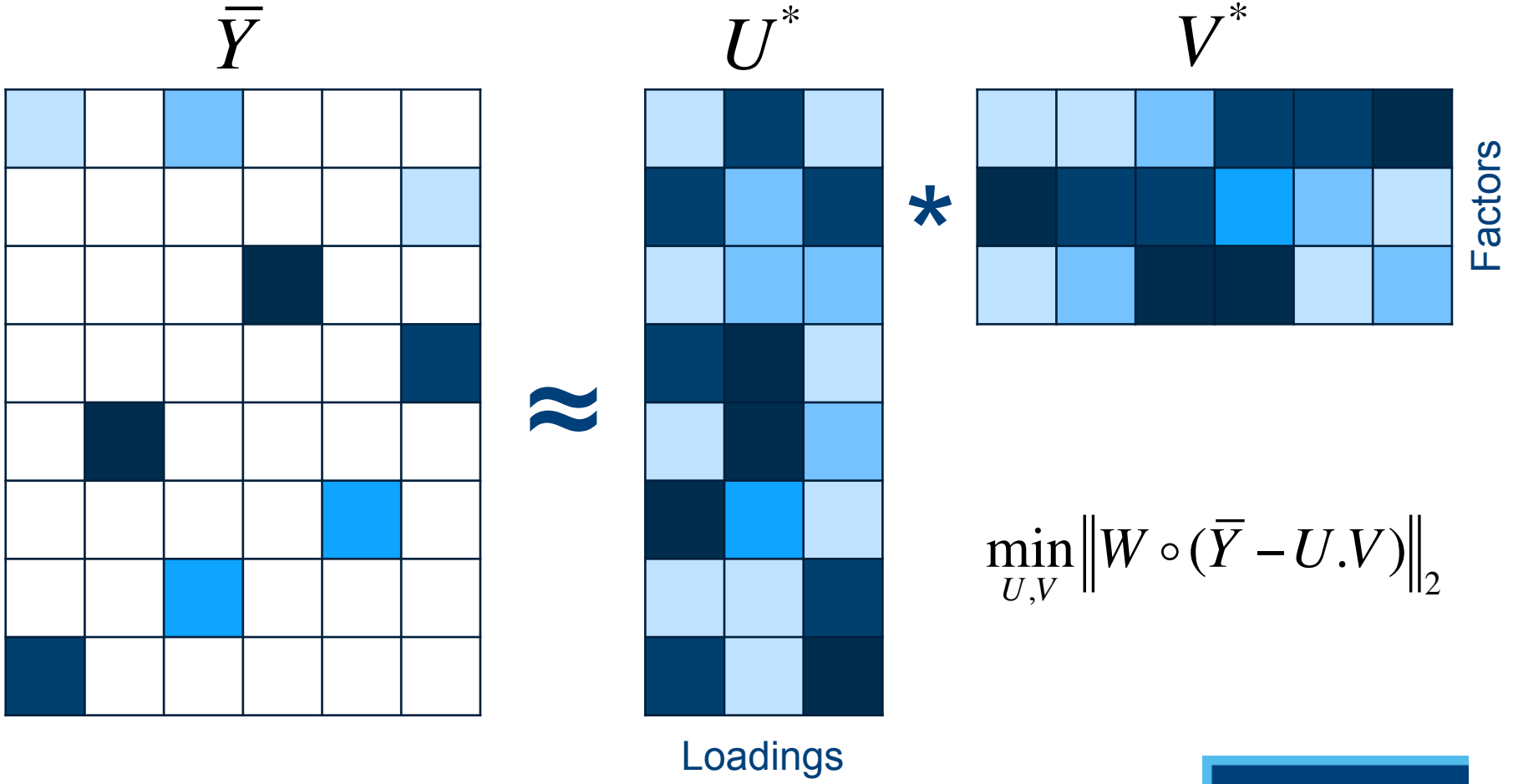
Alternating Least Squares

- If U were known, V could also be found by linear regression



Scarce matrix factorization

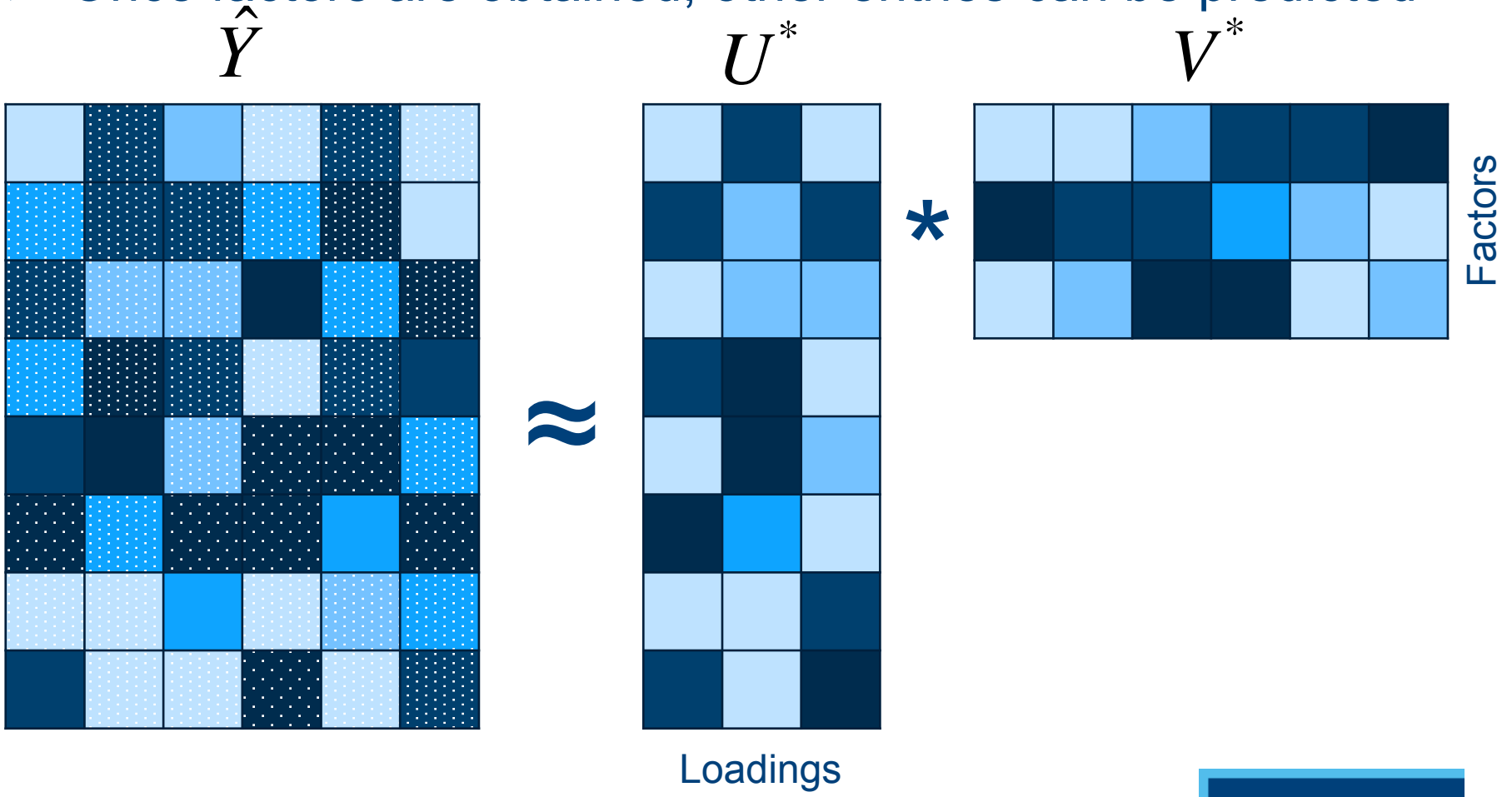
- Only observed values are used in regressions



$$\min_{U,V} \left\| W \circ (\bar{Y} - U.V) \right\|_2$$

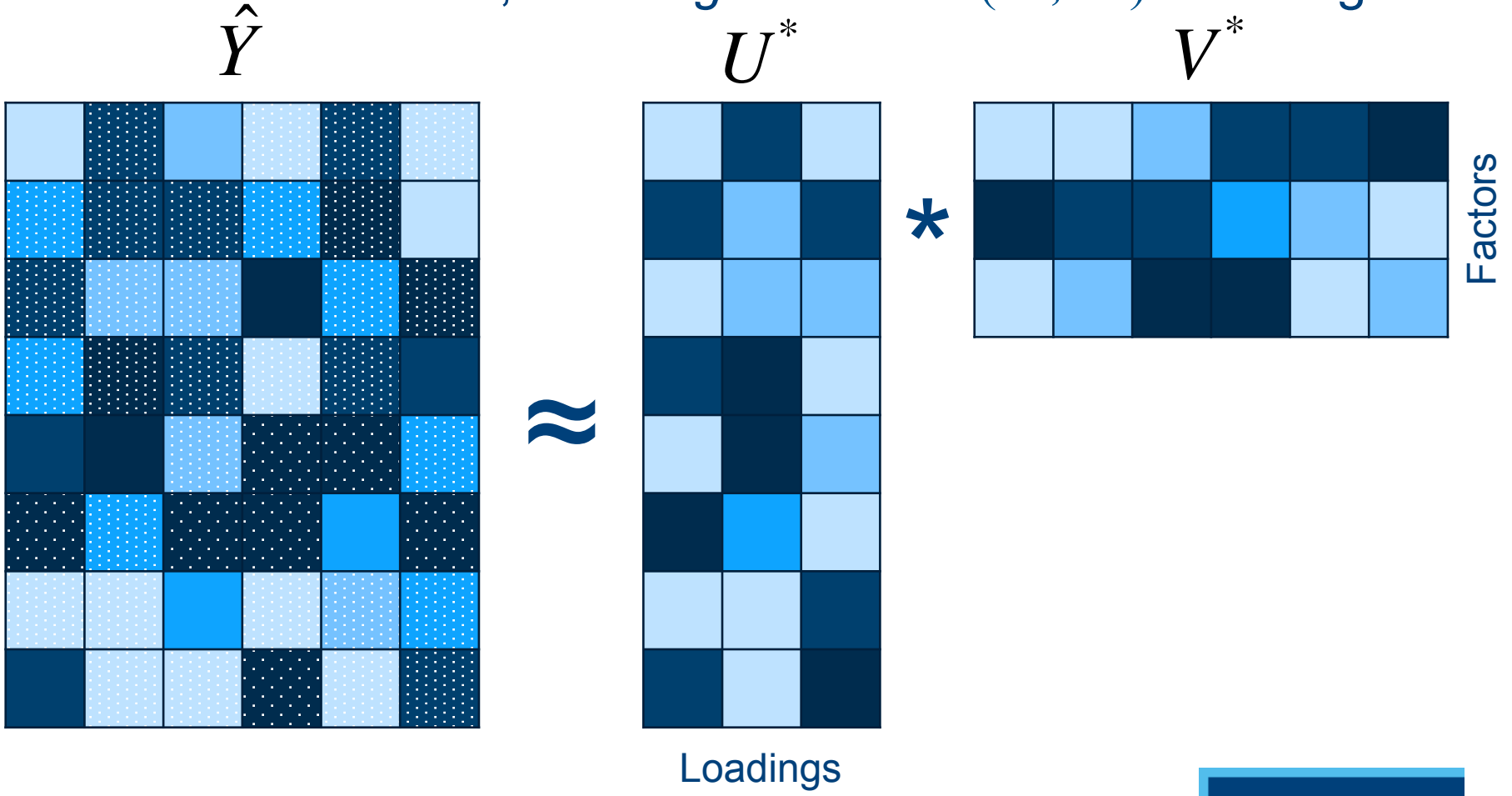
Scarce matrix factorization

- Once factors are obtained, other entries can be predicted



Uncertainty

- Given scarce data, is a *single* solution (U^*, V^*) meaningful?



Bayesian modeling

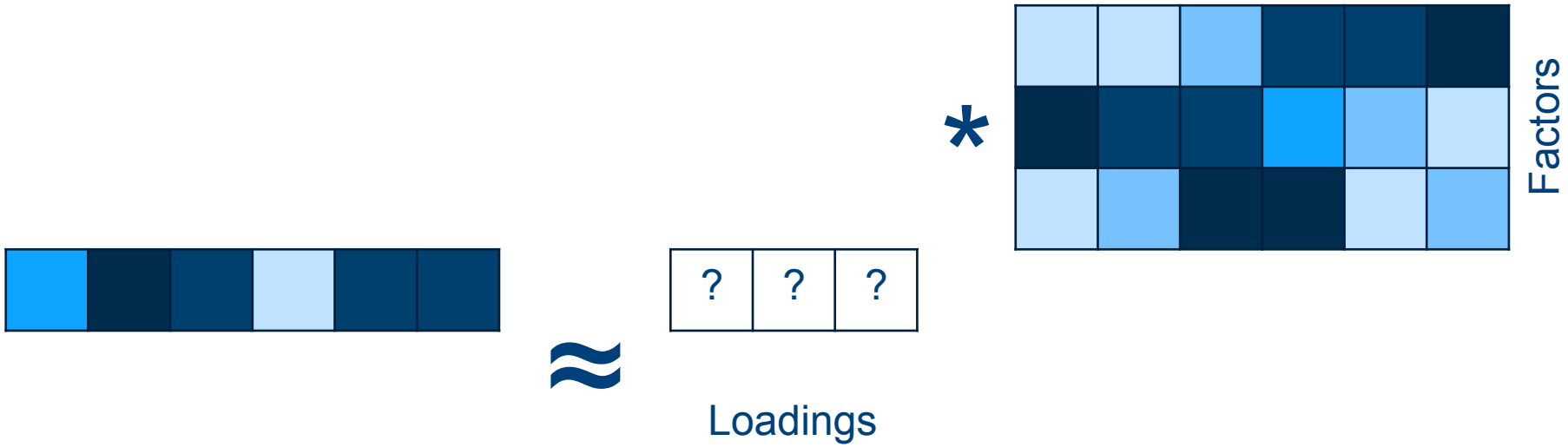
- Given uncertainty from scarce data, Bayesian inference is desirable
  - Instead of $\left(U^*, V^*\right) = \min_{U,V} \left\| W \circ (\bar{Y} - U.V) \right\|_2$, we want to consider the Bayesian posterior distribution
    $$p(U, V \mid \bar{Y})$$
  - Posterior predictive distribution
    $$p(\hat{Y} \mid \bar{Y})$$

    is more informative than any optimal estimator

Ordinary least squares

- ALS involves successive regressions solved by OLS

$Y_{i.}$ $U_{i.}$ V



Ordinary least squares

- Model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} \mathbf{x}_1^{\mathrm{T}} \\ \mathbf{x}_2^{\mathrm{T}} \\ \vdots \\ \mathbf{x}_n^{\mathrm{T}} \end{pmatrix} = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ x_{21} & \cdots & x_{2p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix}, \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix}, \quad \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}.$$

- Solution

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$$

- Setup = transposed of previous notation
- If Gaussian noise, then OLS is max. likelihood estimate

$$\varepsilon \mid X \sim \mathcal{N}(0, \sigma^2 I_n).$$

$$\rho(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma^2) \propto (\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^{\mathrm{T}}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right).$$

Block Gibbs sampler

- The Gibbs sampler is a Markov Chain Monte Carlo method
- MCMC for model inference generates samples from complex posterior distributions of model parameters by iteratively sampling from simpler distributions
- The following scheme is a block Gibbs sampler

$$U^{(i+1)} \sim p(U \mid V^{(i)}, Y)$$

$$V^{(i+1)} \sim p(V \mid U^{(i+1)}, Y)$$

- Under mild conditions of ergodicity, after *burn-in*, the samples will be *dependently* drawn from joint distribution

$$\text{For } i \text{ sufficiently large, } (U^{(i)}, V^{(i)}) \sim p(U, V \mid Y)$$

- Similar to alternating least squares, but global optimization

Markov Chain Monte Carlo

- We do not get the posterior distribution analytically, only samples from it
- Samples are sufficient to characterize posterior distribution
  - *e.g.,* average solutions to get posterior mean estimate
  - *e.g.,* marginal variance of individual predictions to characterize uncertainty

Bayesian linear regression

- The distribution of β in function of the data X and y can be modeled as a multivariate Gaussian distribution over β
- Model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \qquad \varepsilon \mid X \sim \mathcal{N}(0, \sigma^2 I_n).$$

$$\rho(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma^2) \propto (\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^{\mathrm{T}}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right).$$

- Assume a Gaussian prior for β and an inverse gamma prior for ρ

$$\rho(\boldsymbol{\beta}, \sigma^2) = \rho(\sigma^2)\rho(\boldsymbol{\beta}|\sigma^2), \qquad \rho(\sigma^2) \propto (\sigma^2)^{-(v_0/2+1)} \exp\left(-\frac{v_0 s_0^2}{2\sigma^2}\right).$$

$$\rho(\boldsymbol{\beta}|\sigma^2) \propto (\sigma^2)^{-k/2} \exp\left(-\frac{1}{2\sigma^2}(\boldsymbol{\beta} - \boldsymbol{\mu}_0)^{\mathrm{T}}\boldsymbol{\Lambda}_0(\boldsymbol{\beta} - \boldsymbol{\mu}_0)\right) = \mathcal{N}\left(\boldsymbol{\mu}_0, \sigma^2\boldsymbol{\Lambda}_0^{-1}\right).$$

Bayesian linear regression

- Then the posterior distribution of β is also a Gaussian distribution by application of Bayes' rule

$$\rho(\boldsymbol{\beta}, \sigma^2 | \mathbf{y}, \mathbf{X}) \propto \rho(\boldsymbol{\beta} | \sigma^2, \mathbf{y}, \mathbf{X}) \rho(\sigma^2 | \mathbf{y}, \mathbf{X}),$$

$$\rho(\boldsymbol{\beta} | \sigma^2, \mathbf{y}, \mathbf{X}) = \mathcal{N}\left(\boldsymbol{\mu}_n, \sigma^2 \boldsymbol{\Lambda}_n^{-1}\right)$$

$$\boldsymbol{\Lambda}_n = (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \boldsymbol{\Lambda}_0)$$
$$\boldsymbol{\mu}_n = (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \boldsymbol{\Lambda}_0)^{-1}(\boldsymbol{\Lambda}_0\boldsymbol{\mu}_0 + \mathbf{X}^{\mathrm{T}}\mathbf{y}),$$

$$\rho(\sigma^2 | \mathbf{y}, \mathbf{X}), = \text{Inv-Gamma}\left(a_n, b_n\right)$$

$$a_n = a_0 + \frac{n}{2}, \qquad b_n = b_0 + \frac{1}{2}(\mathbf{y}^{\mathrm{T}}\mathbf{y} + \boldsymbol{\mu}_0^{\mathrm{T}}\boldsymbol{\Lambda}_0\boldsymbol{\mu}_0 - \boldsymbol{\mu}_n^{\mathrm{T}}\boldsymbol{\Lambda}_n\boldsymbol{\mu}_n).$$

- *If* $\Lambda_0 = 0$ and $\mu_0 = 0$, then solution for μ_n is identical to OLS!
- Average solution μ_n is similar to ridge regression solution
- Precision matrix Λ_n characterizes variance of solution

GAMBLR trick

- Executing the Gibbs sampler requires sampling repeatedly from posterior Gaussian distributions (which change every time U and V change)
- Sampling from multivariate Gaussian distribution

  $\varepsilon \sim N(0, I)$. If A such that Σ=AA', then $z = \mu + A\varepsilon \sim N(\mu, \Sigma)$
- For Bayesian linear regression

  $$\bar{X} = \begin{bmatrix} X \\ L_0 \end{bmatrix}, \bar{y} = \begin{bmatrix} y \\ L_0\mu_0 \end{bmatrix} \text{ with } \Lambda_0 = L_0 L_0'$$

  $$\mu_n = \left(\bar{X}\bar{X}'\right)^{-1} \bar{X}\bar{y} \text{ and } \Lambda_n = \bar{X}\bar{X}'$$

  It can be shown that $z = \left(\bar{X}\bar{X}'\right)^{-1} \bar{X}(\bar{y} + \sigma.\varepsilon) \sim N(\mu_n, \sigma^2\Lambda_n^{-1})$
- This has the same form as OLS!

GAMBLR trick

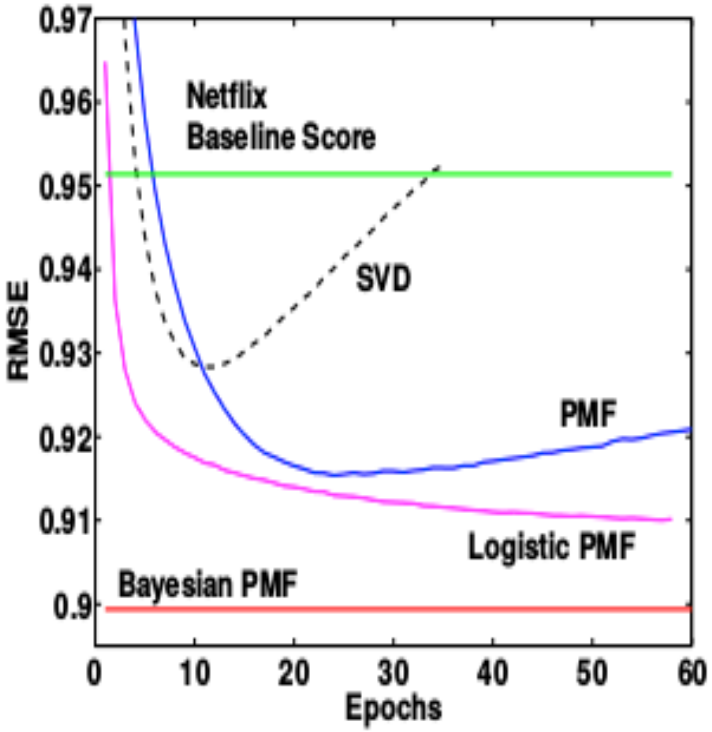
- This means that we can sample from the posterior Gaussian distribution by solving a linear regression on the original data plus injected noise!
- Running the Gibbs sampler then only amounts to solving a sequence of linear regressions with variable noise injection!
- Linear regression is one of the best studied problems in numerical analysis
  - Fast algorithms
  - Scalable code
  - One multivariate regression per row or column of Y at each iteration step, hence easy parallelization

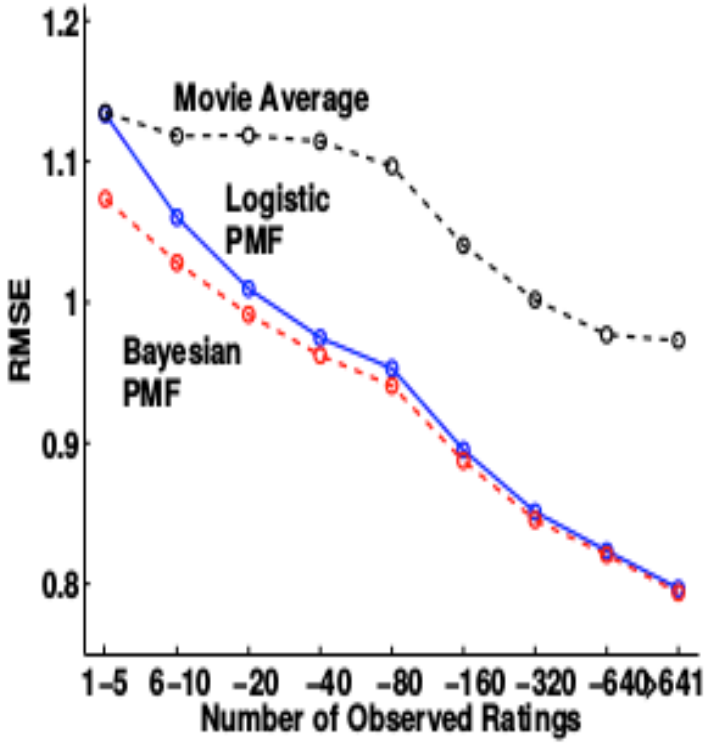
Matrix factorization

- One of the best approaches for Netflix challenge
  - Prediction of ratings for viewer-movie pairs
- ***Does not use features, only matrix values***
- Two popular versions
  - Probabilistic Matrix Factorization (PMF) = Maximum Likelihood
  - Bayesian PMF = Bayesian inference

Netflix comparison (PMF vs. BPMF)

- Data: 100M ratings from 480K users, 18K movies
- BPMF has advantage for users with few ratings

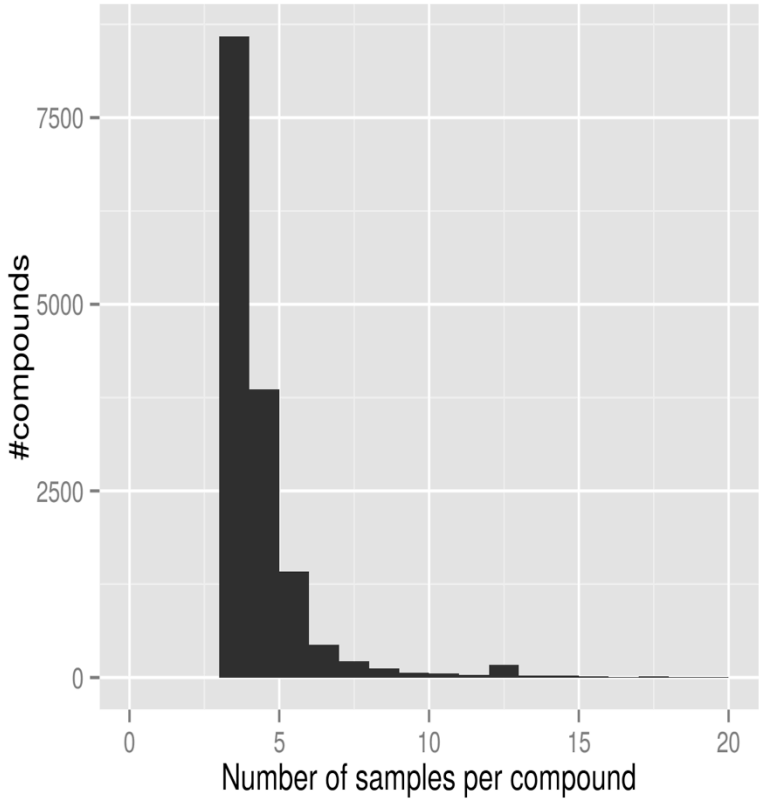




KU LEUVEN

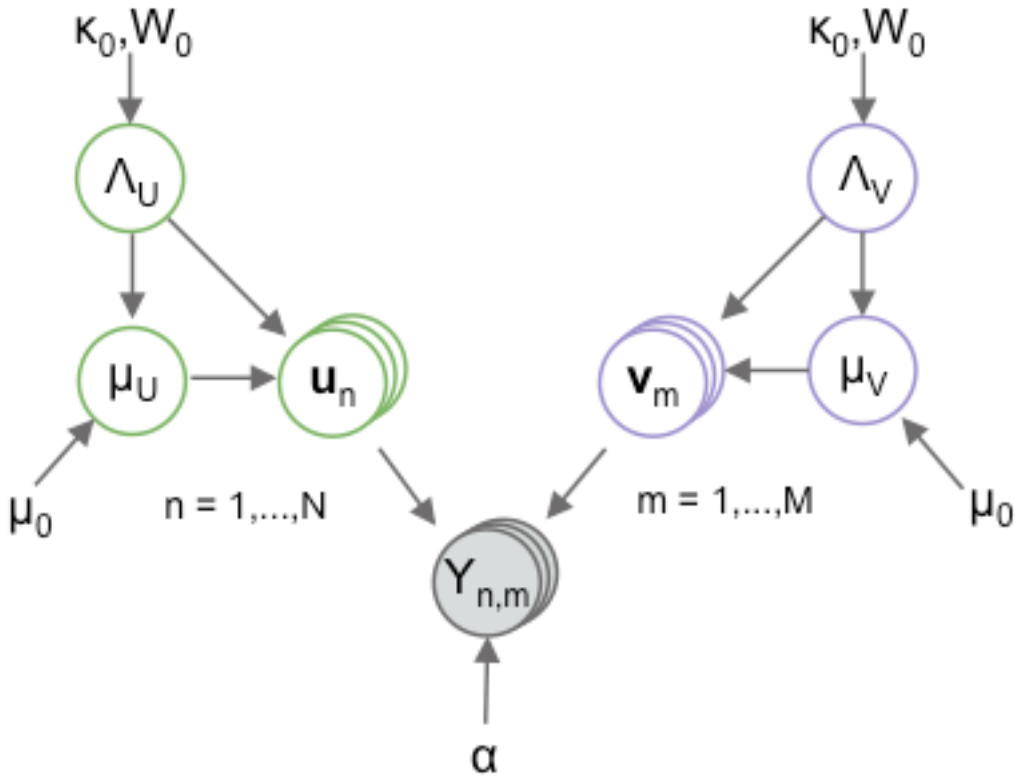
Motivation for Bayesian PMF

- PMF gives point estimates
  - Problematic for compounds that have only **few samples**
  - We are interested in **uncertainty** of estimates



Example IC50 data set from CHEMBL with 15K compounds

Bayesian PMF



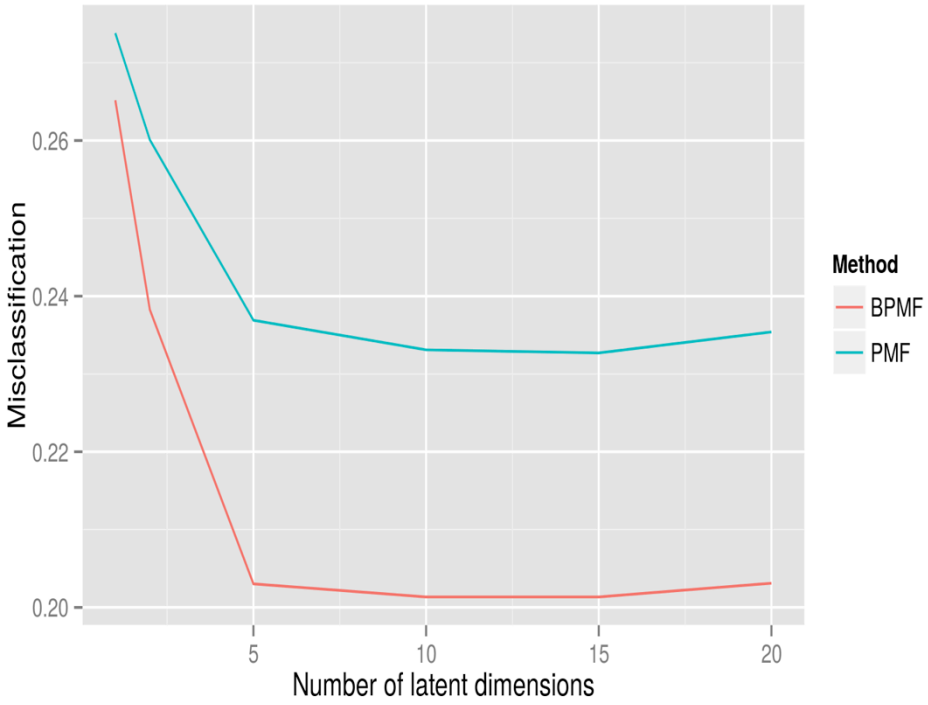
Gibbs sampling

- Iteratively samples each parameter
- Obtains posterior samples of the model
  - *e.g.*, sample 200 models after burn-in
- Using the samples one can also measure uncertainty
- Related to Alternating Least Squares
- Blocked Gibbs sampler with large blocks, good sampling behavior

ChEMBL: PMF vs. Bayesian PMF

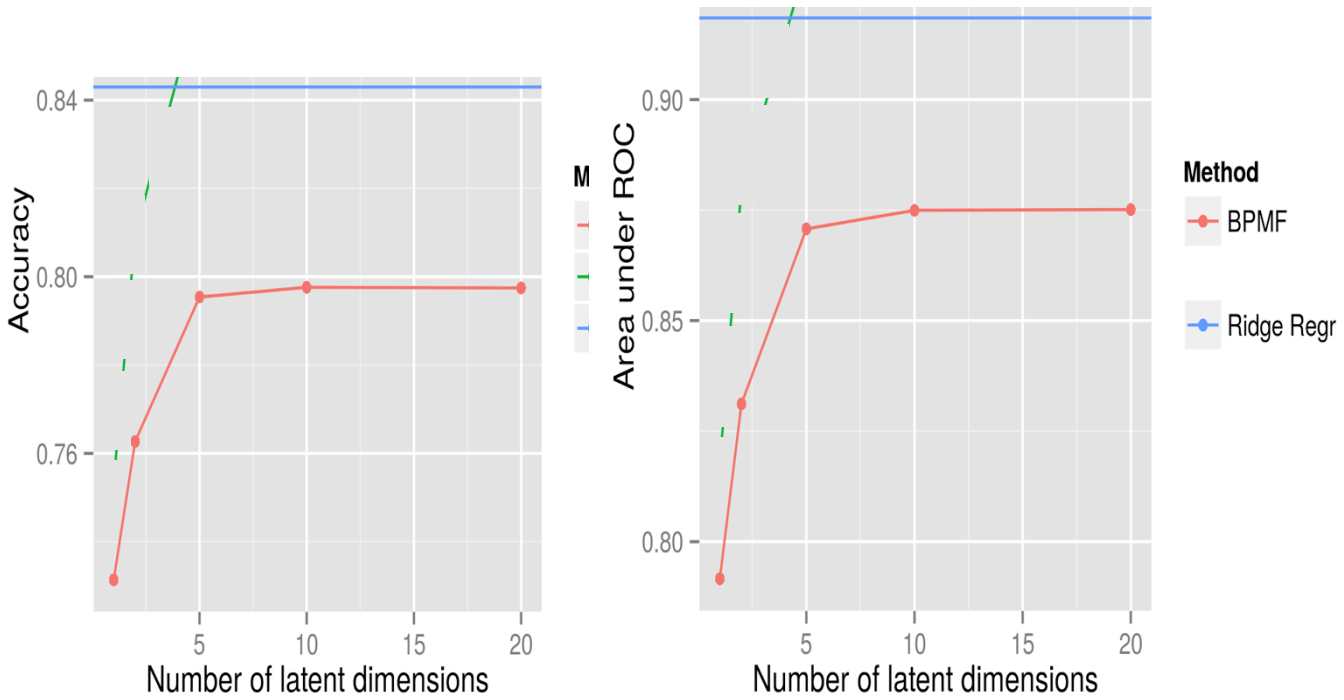
- ChEMBL public data set of assay activities
- Classified IC50
  - 15,118 compounds
  - 344 proteins
  - 59,451 values
  - Discretization at 200nM
  - 20% test
- BPMF outperforms PMF
- ***Does not use features, only matrix values***

Test classification error



KU LEUVEN

ChEMBL: BPMF vs. ridge regression



15K compounds
344 protein
200 nM threshold

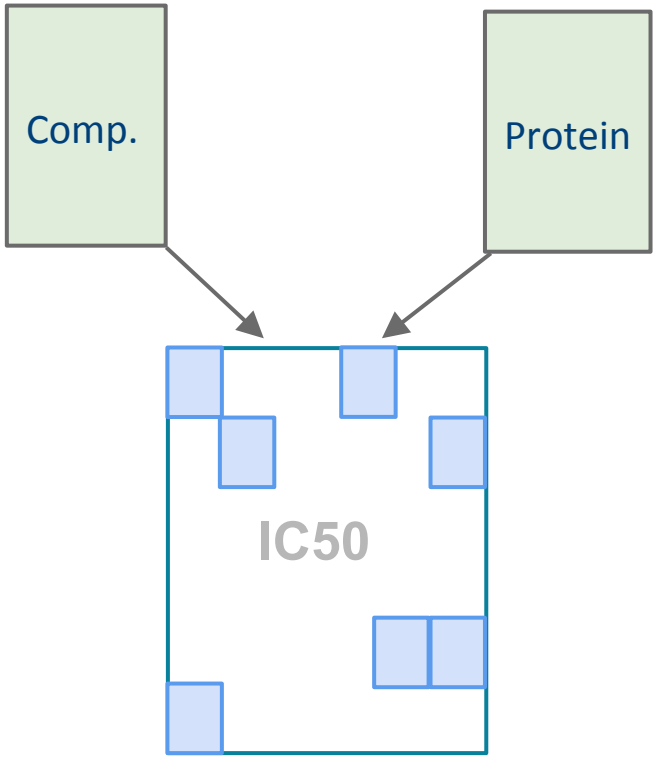
20% for test set

Vary number of dimensions

Matrix factorization not as good as QSAR, but does capture information.

KU LEUVEN

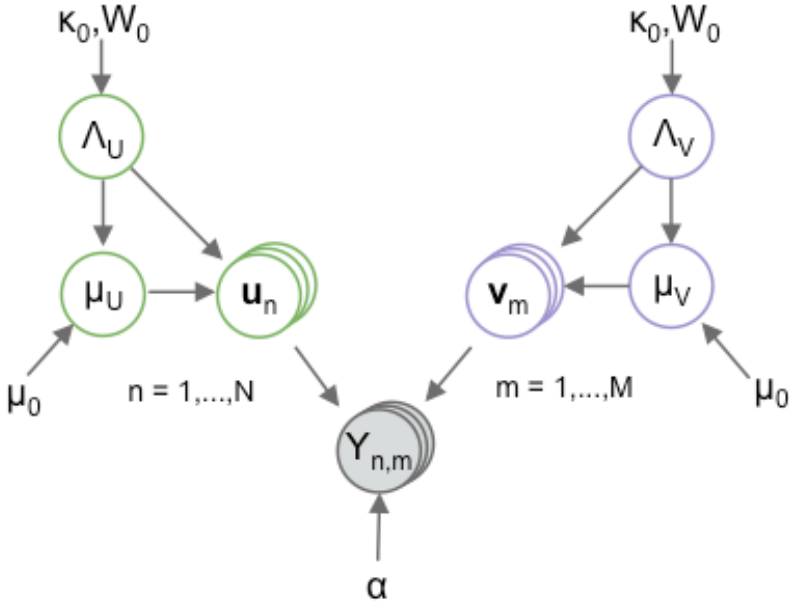
BPMF (relation view)



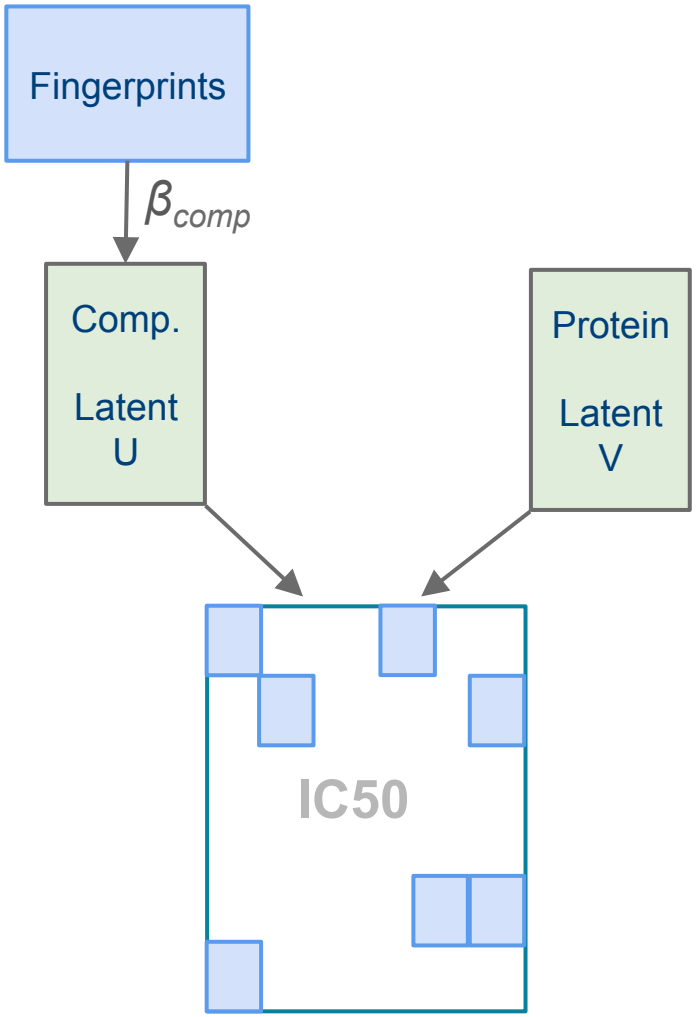
Model

2 entities, 1 relation

Latent variables (green) are learned from the **IC50** data.



Macau



Can we get
the best of both worlds?

Model

2 entities, 1 relation
+ features for compounds

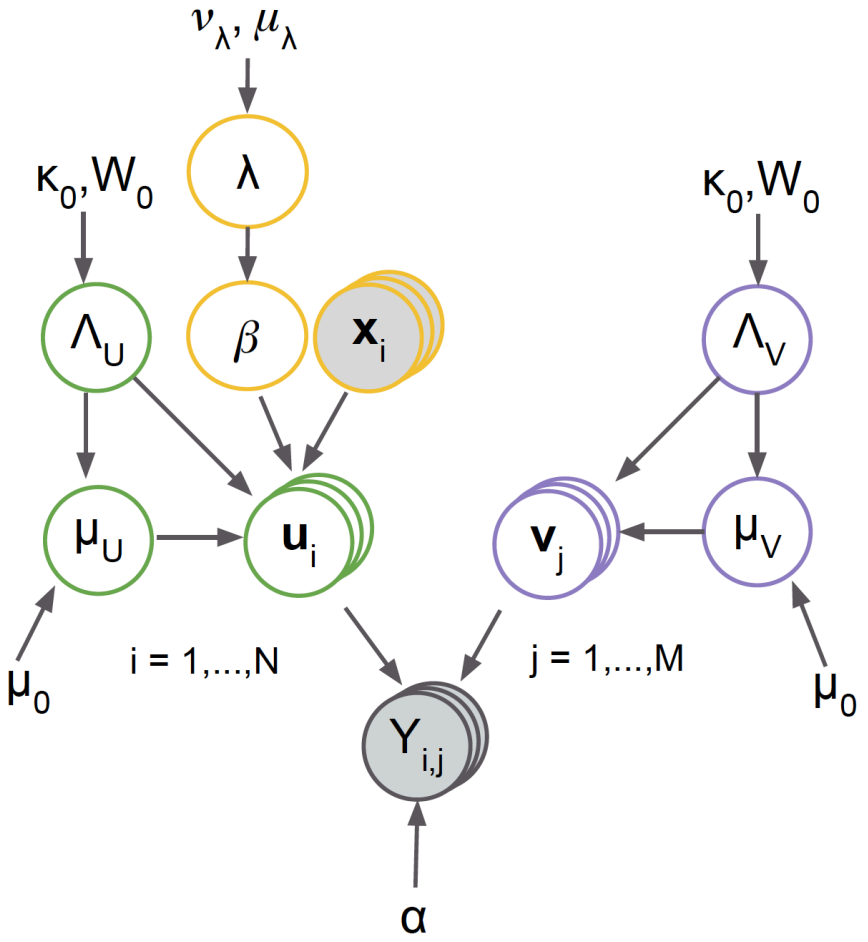
Latent variables are learned
together with β_{comp}

Using side information

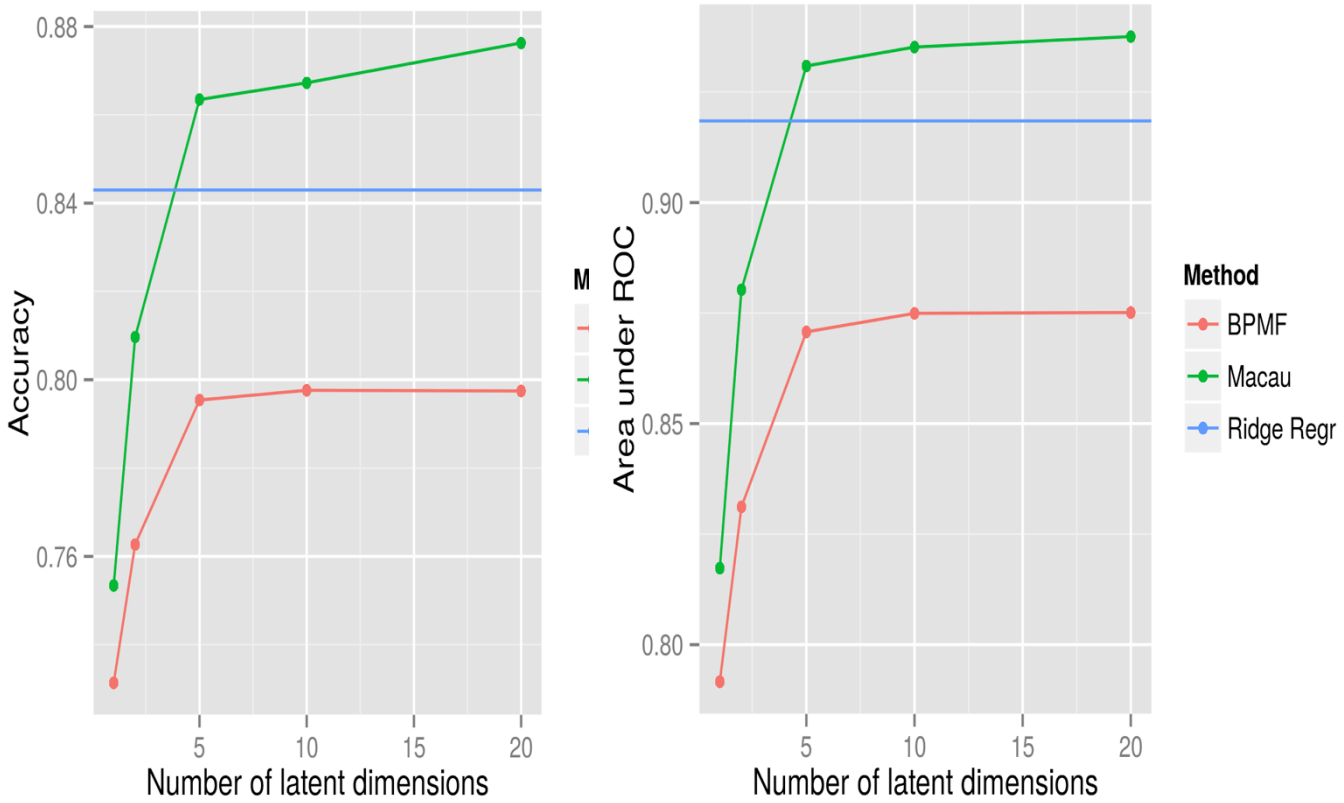
- We incorporate side information to prior mean of latent vectors

$$\mathbf{u}_i \sim \mathcal{N}(\mu_U + \beta^\top \mathbf{x}_i, \Lambda_U)$$

  - $\mathbf{x}_i$ is feature vector
  - β is link matrix
- β and λ (precision) are also learned



Results on ChEMBL



15K compounds

344 protein

200 nM threshold

20% for test set

Compound features improve performance

Multitask modeling improves performance

KU LEUVEN

Sampling the link matrix (1)

- β is **F x D** matrix, where
  - F (the number of features) can be bigger than 100k or 1M.
  - D is the number of latent dimensions
- Conditional posterior of β is

$$p(\beta) \propto \exp(-\mathrm{tr}\left\{((\mathbf{U}-\mathbf{X}\beta)^\top(\mathbf{U}-\mathbf{X}\beta) + \lambda\beta\beta^\top)\Lambda_U\right\})$$

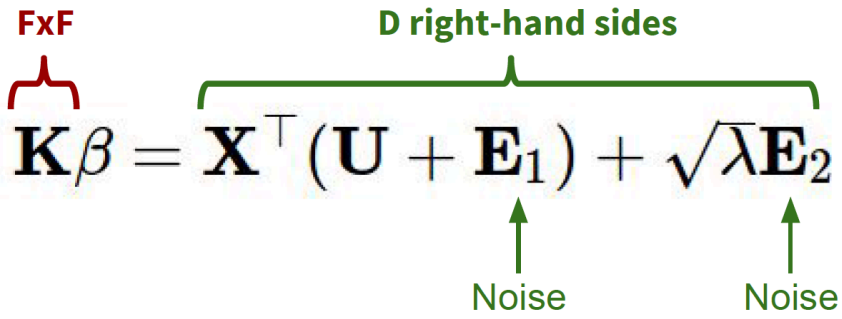
fitting error | prior | scaling

- The chosen prior allowed us to factorize out Λ_U

Noise injection sampler

- Sample of β can be generated by solving linear system:

$$\mathbf{K} = \mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}$$



- Every row in $\mathbf{E}_1$ and $\mathbf{E}_2$ is sampled from $\mathcal{N}(0, \Lambda_U^{-1})$

Industrial scaling (J&J data)

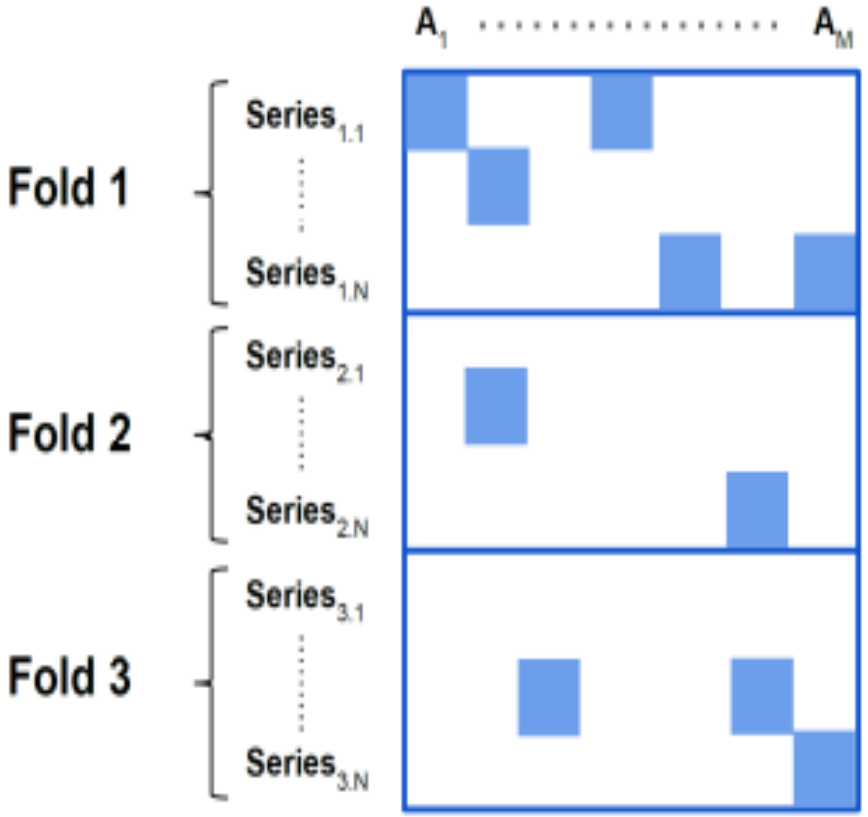
- ~2M compounds, ~1K targets, tens of millions of activities
  - Compute nodes: dual Xeon E5-2699 v3
  - Fingerprint 1: 6,000 features
    - Latent dimension = 30
    - Direct solver on single node
      - 40s per Gibbs sampling pass
      - 1,000 iterations (800 burn-in) = ½ day
  - Fingerprint 2: 4,000,000 features
    - Sparsity of X: 0.002%
    - Latent dimension = 30
    - Iterative solver on 15 nodes
      - 600s per Gibbs sampling pass
      - 1,000 iterations (800 burn-in) = 1 week

Single-task vs. multitask learning

- SVM using scikit-learn
  - Separate classifier for every assay
  - Hyperparameter by nested CV
    - For each assay separately
  - Linear kernel
    - Gaussian kernel has equivalent performance but does not scale
- Macau classification using TensorFlow
  - Non-Bayesian approach (optimization)
  - Multi-task learning
  - Hidden representation size: 1,000
  - Model parameters chosen by ChEMBL experiments

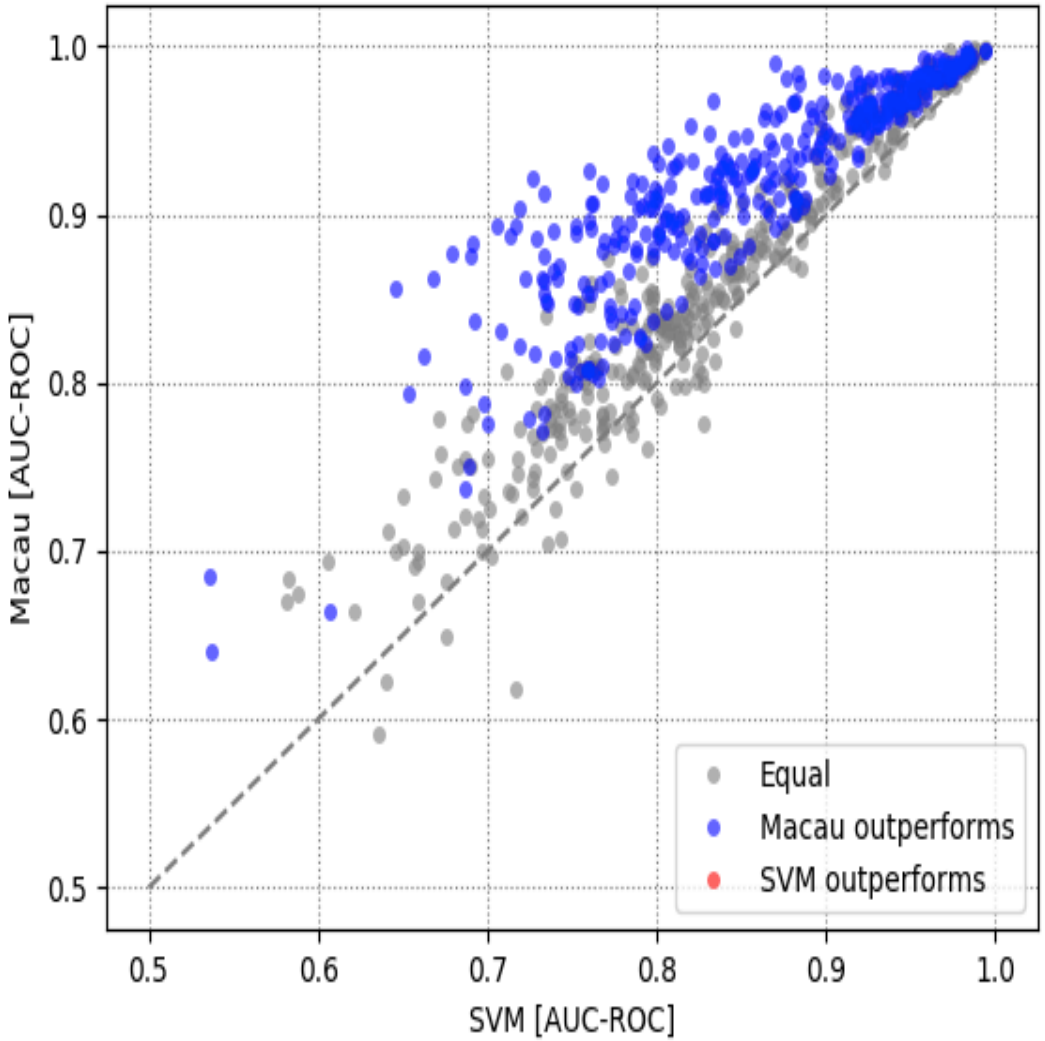
Nested clustered crossvalidation

- Chemical series effect
  - All members of a series should be either in training or test set
- Clustering
  - Tanimoto > 0.7
- Nested cross-validation for hyperparameter tuning



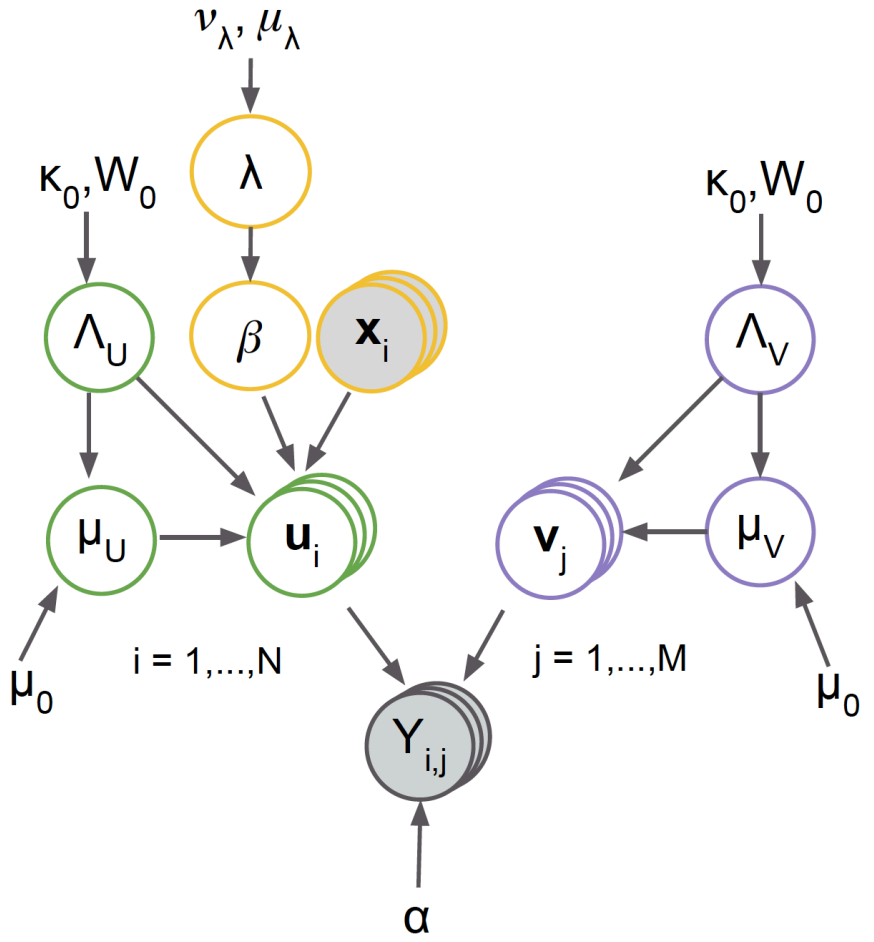
AUC per assay

- Mean over assays
  - Macau: **0.886**
  - SVM: **0.840**
- From **712** assay
  - Macau wins **382**
  - SVM wins **0**
  - Ties **330**
  - Using $p < 0.01$



Variational Bayes

- Gibbs sampling = “old”
- Variational Bayes popular
- Hierarchical blindness in VB
  - Ignored covariance between β and latents **u**
  - Poor variance estimates
- u_i covariance increases if side information



KU LEUVEN

Empirical comparison: ChEMBL

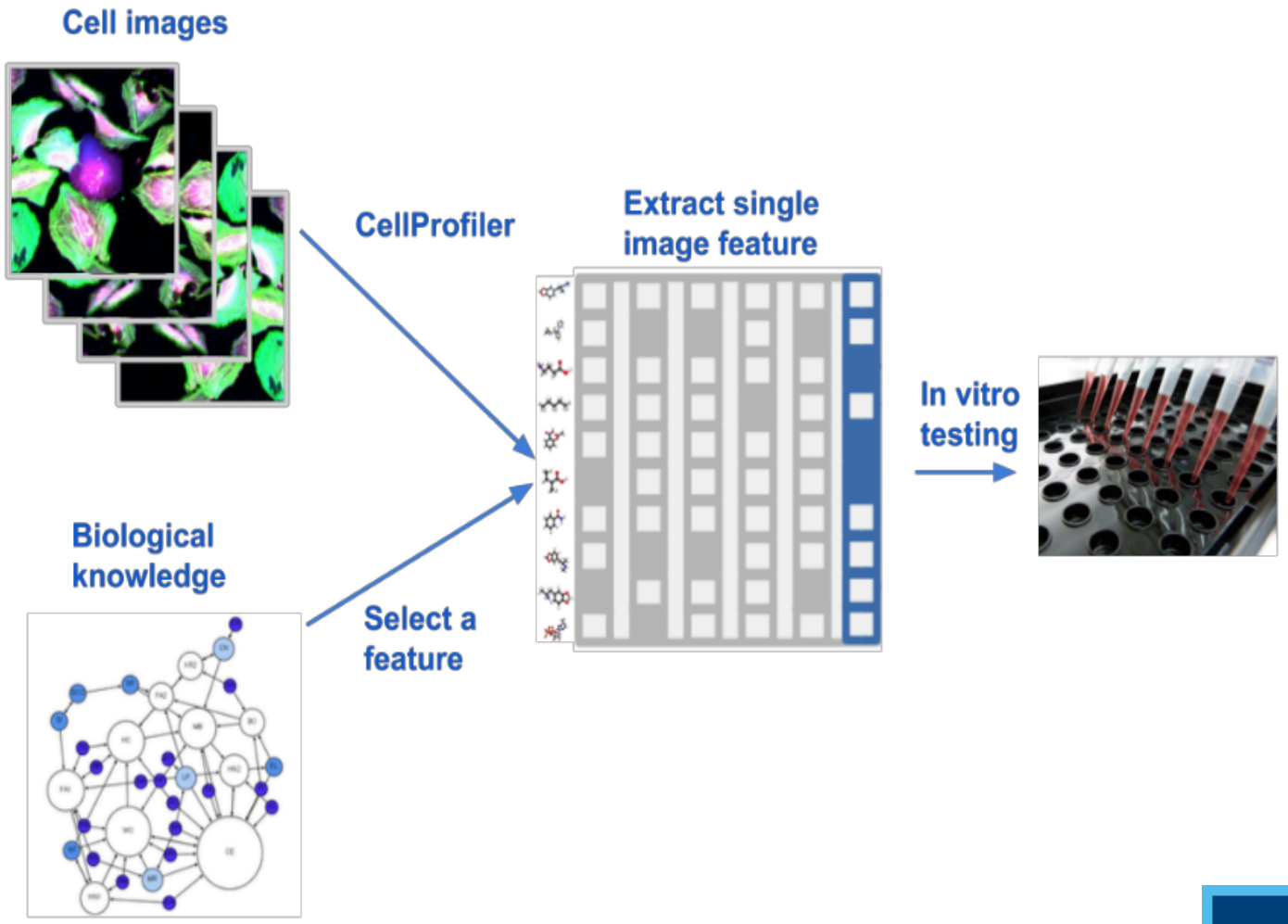
- 15k compounds
- 346 proteins
- ~60k activity measurements
  - pIC50
  - 20% test set
- Sparse high-dimensional **side information** (#feat is ~100k)
- Macau drastically outperforms VBMFSI

METHOD	RMSE	NEGLL
BMF(MCMC)	0.8948 (0.0072)	1.2252 (0.0078)
BMF(VB)	1.0045 (0.0057)	1.3933 (0.0048)
LIBFM	0.6510 (0.0072)	-
VBMFSI-CA	0.8024 (0.0111)	1.1678 (0.0141)
MACAU (OURS)	**0.6122** (0.0053)	**0.8756** (0.0050)
VAFFL (OURS)	0.6829 (0.0080)	1.0091 (0.0110)

Repurposing High-Content Imaging data

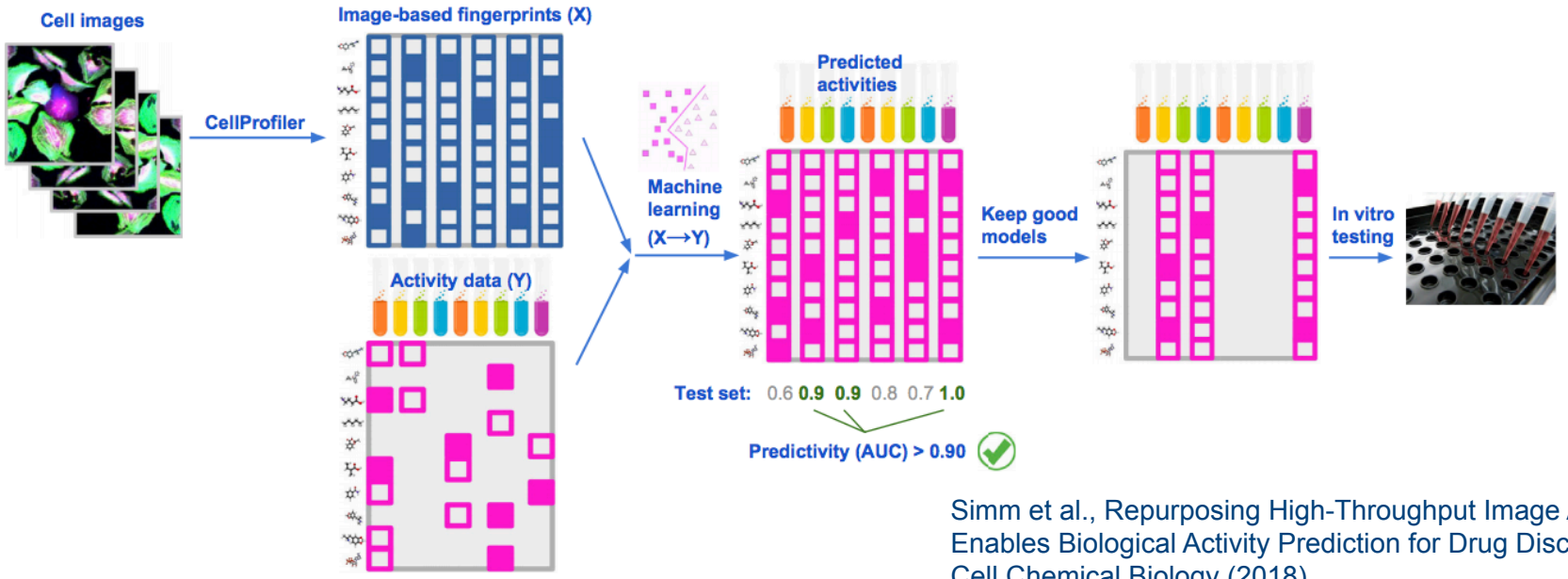
KU LEUVEN

Classical high-content imaging



Repurposing imaging assays

- High-throughput imaging (= high-content screening)
- 500K compounds, 600 drug targets, 10M activities (30% fill rate)
- Glucocorticoid receptor assay phenotypic screen
  - Feature extraction from images with CellProfiler



Simm et al., Repurposing High-Throughput Image Assays Enables Biological Activity Prediction for Drug Discovery, Cell Chemical Biology (2018)

Figure 2. Strategy to Repurpose Imaging Screens to Efficiently Predict Biological Activity

Features extracted from images of cells are used by machine-learning methods to model all available activity data from previously performed assays. Assays with good predictivity on the test data are then selected for testing a relatively small number of predicted-active compounds, chosen from a large set of compounds profiled in the imaging assay.

Application

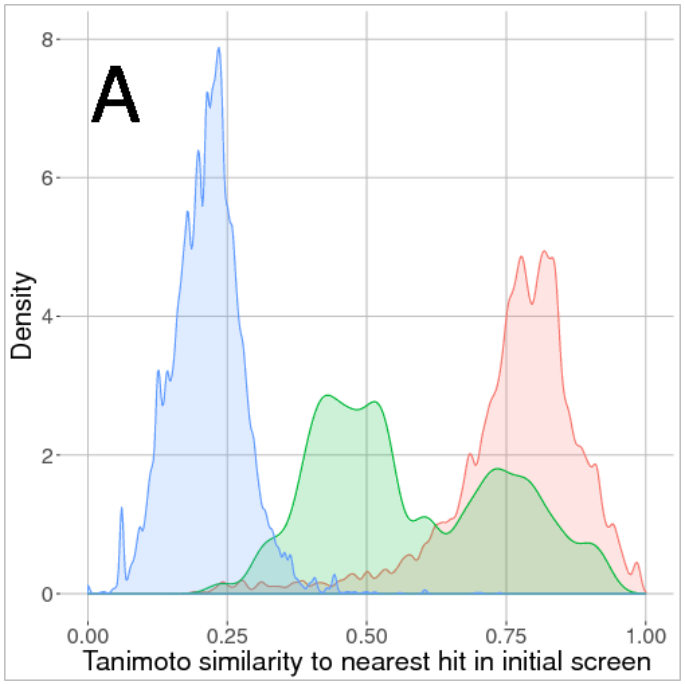
- Oncology drug discovery project
  - Active project
    - Initial screen = 0.725% hit rate (submicromolar)
  - Kinase target
  - No known direct relation to glucocorticoid receptor
  - Rank unscreened compounds with imaging data
  - Test top 342 compounds
    - 141 submicromolar hits (41% hit rate)
    - 60x enrichment

Application

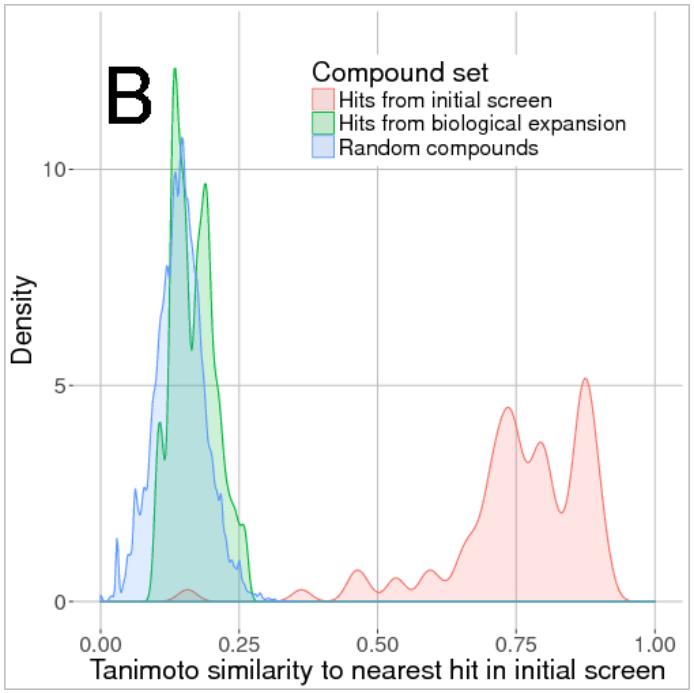
- Central nervous system project
  - Active project
    - Initial screen = 0.088% hit rate
  - Enzyme target
  - No known direct relation to glucocorticoid receptor
  - Rank unscreened compounds with imaging data
  - Some additional ADME filtering
  - Select 141 compounds
    - 37 submicromolar hits (22.7% hit rate)
    - x250 enrichment

Imaging data improves chemical diversity

- Similar or better hit rates using structure fingerprints
- BUT high chemical diversity (biologically driven vs. chemically driven)



Oncology



CNS

KU LEUVEN

Imaging assays for drug discovery

- 500K compounds, 600 targets, 10M activities (30% fill rate)
- Glucocorticoid receptor assay phenotypic screen

- Evaluate predictivity using clustered cross-validation

- Macau predictive for 37% of assays (CV AUC>0.7), highly predictive for 5% of assays (CV AUC>0.9)
  - Assays not related to original screen!

- Here: single imaging assay
- Future: build systematic library of imaging assays

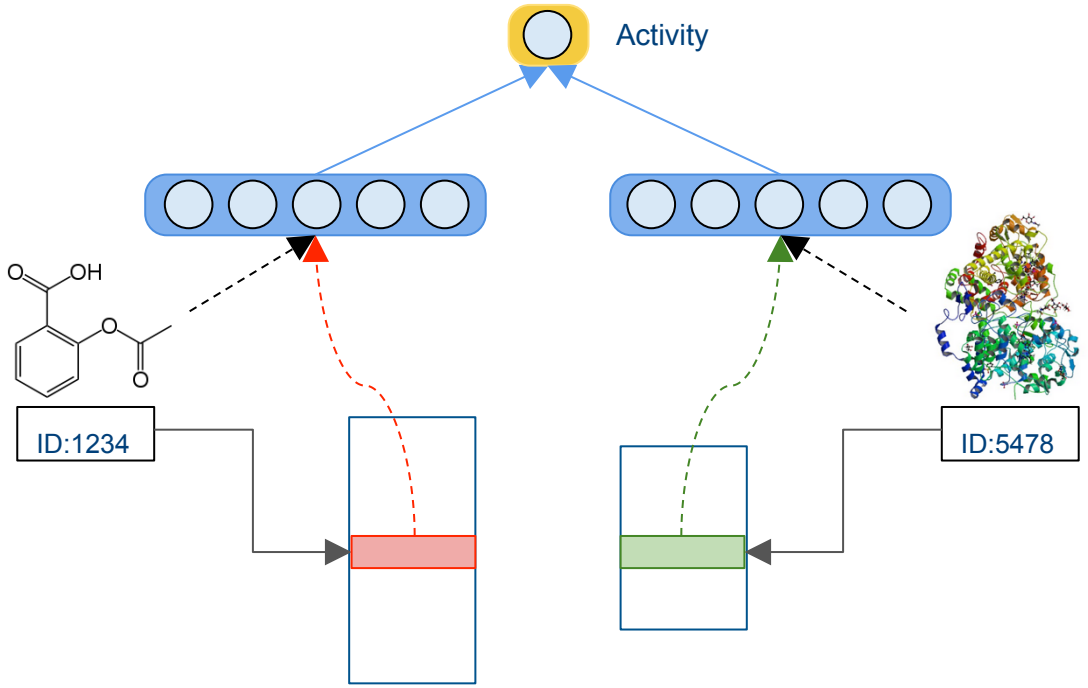
Macau

- Generic package
- Open source

- OpenMP/C++ with Python wrapper library
  - https://github.com/jaak-s/macau
  - Factorization with and without side information
  - Real valued and binary matrices (normal and probit noise)
  - Supports tensors (alpha)
  - Univariate and multivariate Gibbs sampler

Deep Macau

- Combine deep learning and matrix factorization
  - Deep learning allows to capture nonlinear effects
  - Matrix factorization allows item level reasoning
  - Instead of only transforming features into prediction, learn a latent representation of each entity

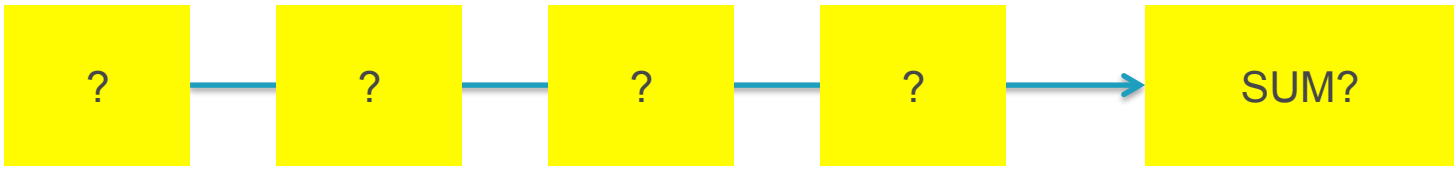


Privacy-Preserving Machine Learning

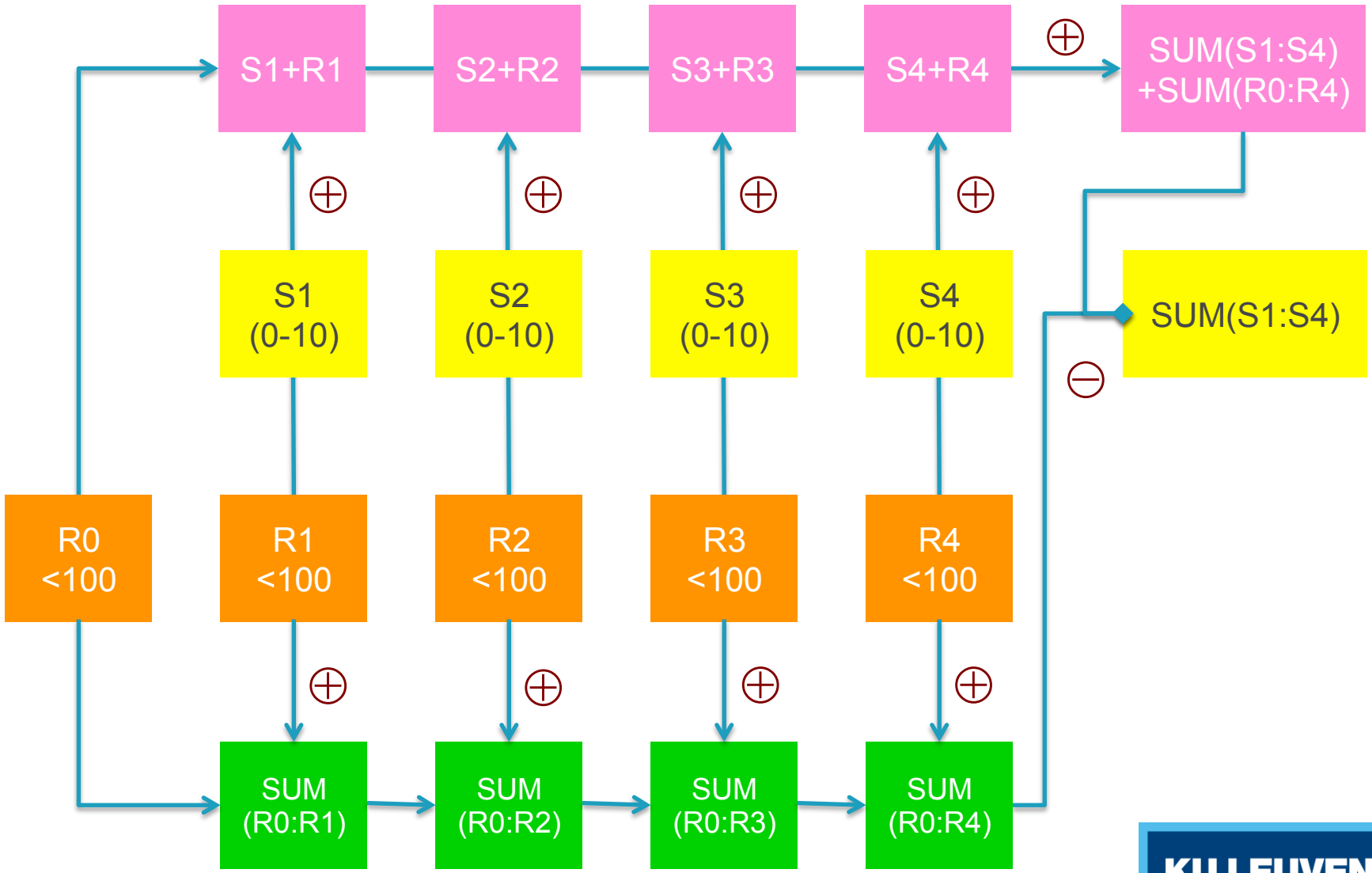
Privacy-preserving modeling

- Partners want to model data jointly across multiple partners
- The partners DO NOT want to disclose the original data to each other
- The partners are willing to disclose some derived data
- How can you model data jointly without disclosing it?!?
  - Privacy-preserving modeling

Privacy-preserving sum



Privacy-preserving sum



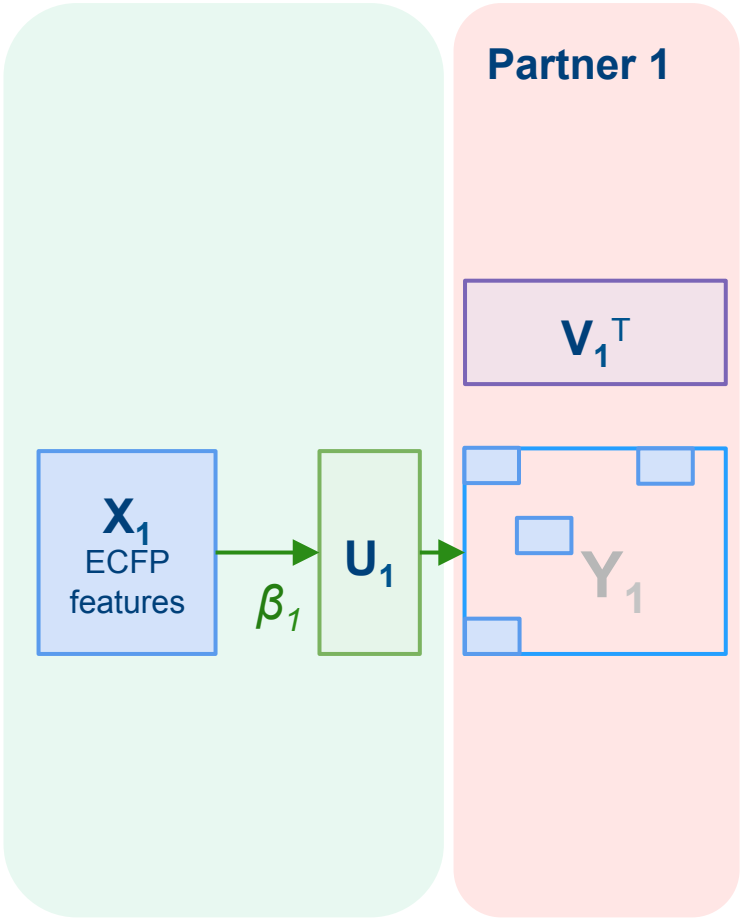
KU LEUVEN

Privacy-preserving sum

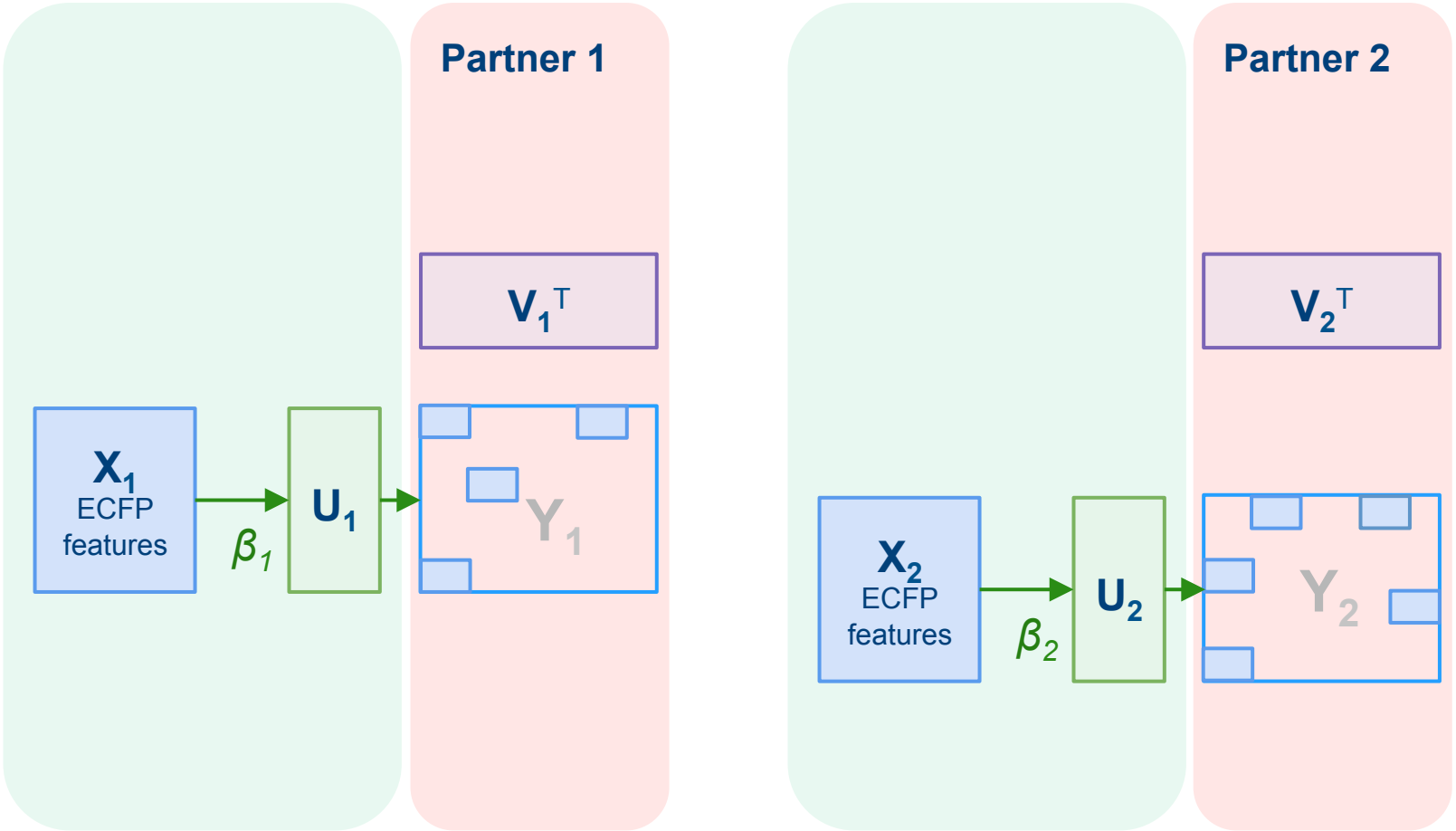
- What we are calculating

$$
\begin{array}{l}
R0+(S1+R1)+(S2+R2)+(S3+R3)+(S4+R4) \\
-((((R0 \quad +R1) \quad +R2) \quad +R3) \quad +R4) \\
\hline
\quad S1 \quad + \; S2 \quad + \; S3 \quad + \; S4
\end{array}
$$

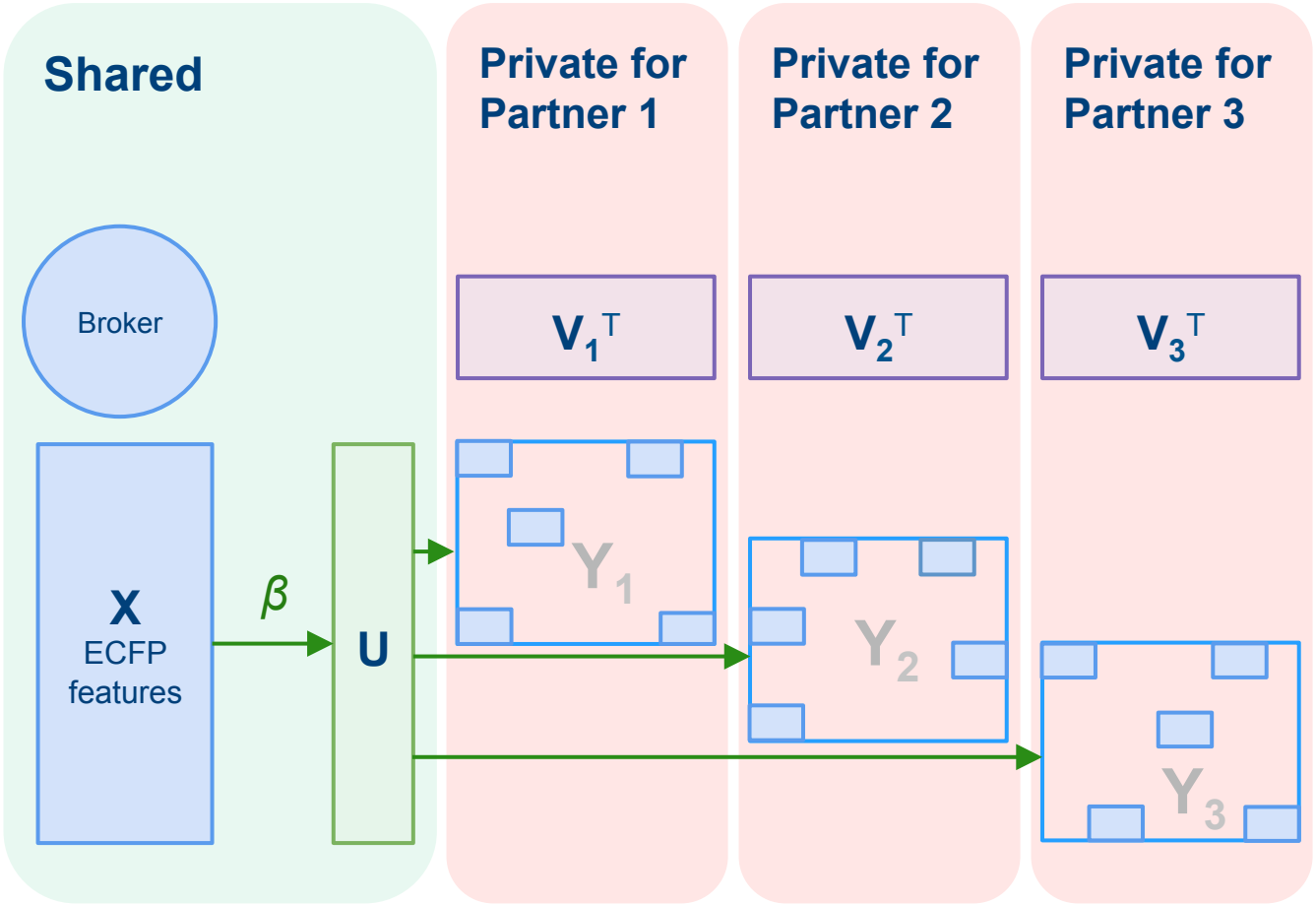
Single-party Macau



Independent parties



Privacy-preserving broker



Initialization
Broker receives X from each partner and aligns them

Iteration

1. Partners privately update **V**
2. Partners send contributions for **U** to broker
3. Broker computes and shares **U**
4. Broker updates ***β***

MachinE Learning Ledger Orchestration for Drug DiscoverY

MELLODDY

Innovative Medicines Initiative
10 pharma partners
€18,000,000
June 2019 – May 2022

PHARMA PARTNERS

AMGEN, astellas, AstraZeneca, BAYER, Boehringer Ingelheim, gsk, janssen, MERCK, NOVARTIS, SERVIER

PUBLIC PARTNERS

MŰEGYETEM 1782, IKTOS, KU LEUVEN, loodse, NVIDIA, OWKIN, SUBSTRA FOUNDATION

This project has received funding from the Innovative Medicines Initiative 2 Joint Undertaking under grant agreement N° 831472. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and EFPIA

imi innovative medicines initiative, efpia

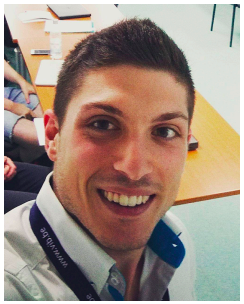
KU LEUVEN

Conclusions

- Fully Bayesian matrix factorization with side information
  - Multitask learning with tasks tied by matrix factorization
- Scalable, parallelizable full MCMC
- Particularly attractive when
  - Modeling prediction uncertainty
  - Scarce target matrix
  - Sparse feature matrix
- State-of-the-art performance on chemogenomic tasks

Jaak Simm

Adam Arany



Daniele Parisi

Pooya Zakeri

Edward De Brouwer

janssen

Hugo Ceulemans
Jörg Wegner

You?

- Postdoc/PhD
- Deep learning
- Privacy-preserving ML
- Chemoinformatics
- EHR

KU LEUVEN