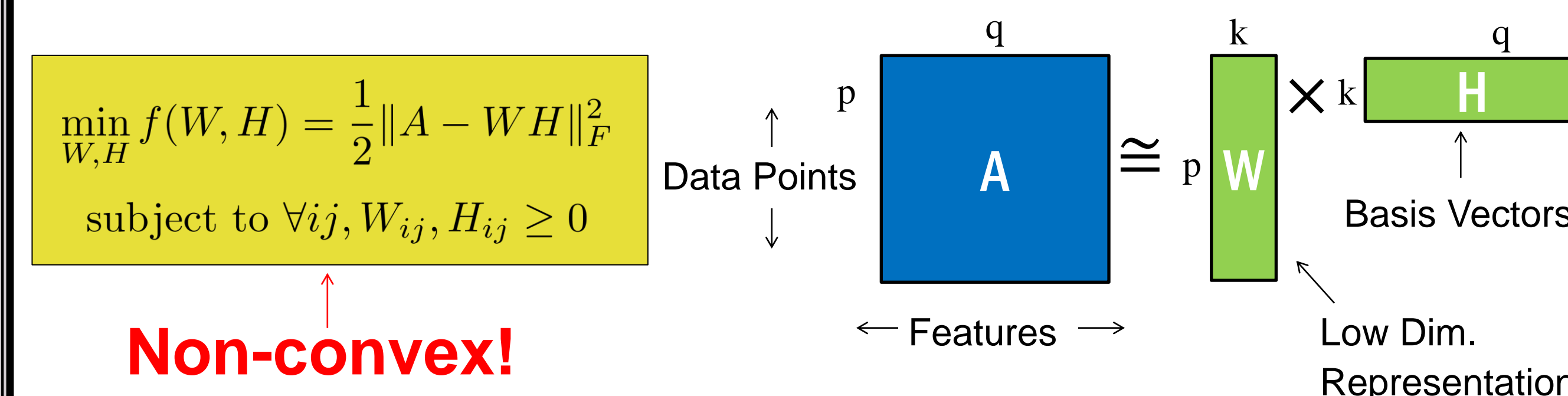


Summary

Non-Negative Matrix Factorization (NMF) is a dimensionality reduction technique that is useful for obtaining a “sum-of-parts” decomposition of data. We describe an optimization method that utilizes the stochastic nature of the data to highly speed up a state of the art NMF algorithm: the Block Principal Pivoting (BPP) method.

Non-Negative Matrix Factorization



Alternating Non-Negative Least Squares (ANLS):

$$\min_{H \geq 0} \|W \times H - A\|_F^2 \quad \min_{W \geq 0} \|H^T \times W^T - A^T\|_F^2$$

Break down as Non-Negative Least Squares (NNLS) problems

$$\min_{x \geq 0} \|C \times x - b\|_2^2$$

Convex! Find x satisfying KKT conditions

$$y = C^T C x - C^T b$$

$$y \geq 0$$

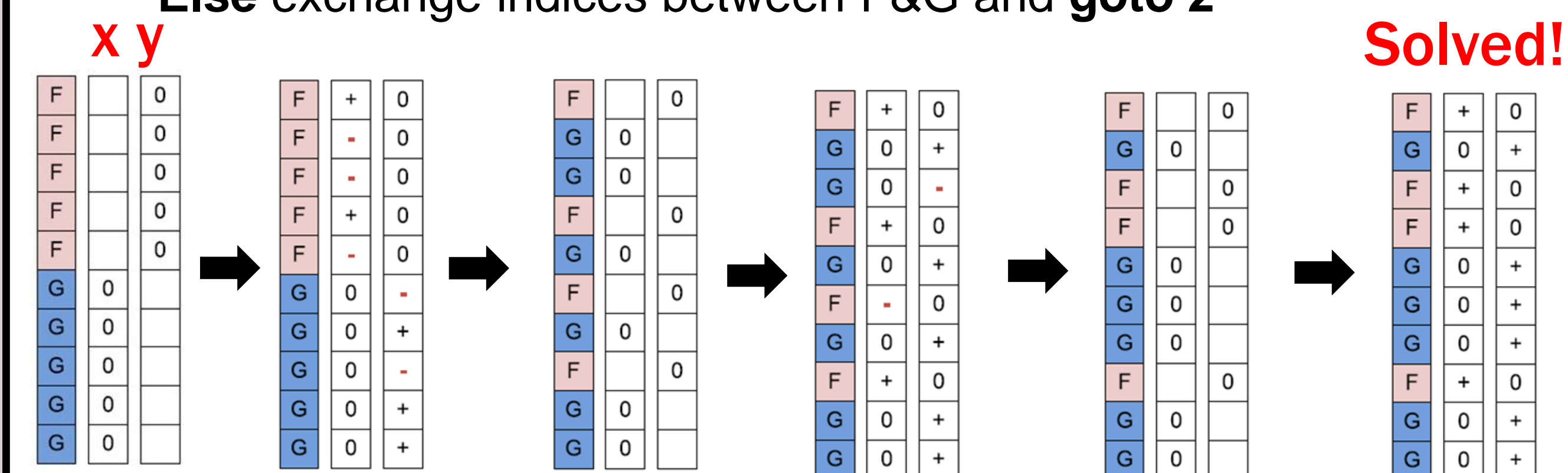
$$x \geq 0$$

$$x_i y_i = 0, i = 1, \dots, d$$

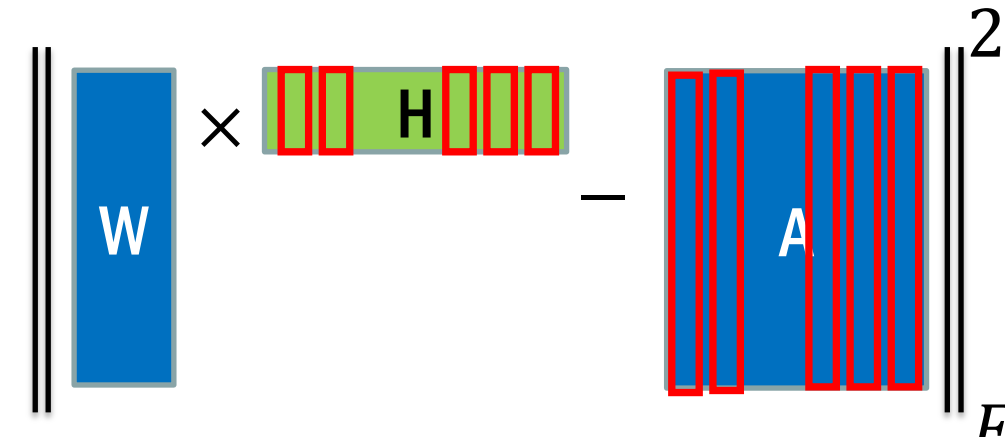
BPP method for NNLS problems

Kim & Park (2008)

1. Partition index set $[1 \dots d]$ into sets F & G
 2. Set $x_G, y_F = 0$ (for satisfying complementary slackness)
 3. Solve $x_F = (C_F^T C_F)^{-1} (C_F^T b)$ and $y_G = (C_G^T C_G) x_F - C_G^T b$
 4. If $x_F \geq 0$ and $y_G \geq 0$, then $x = (x_F, x_G)$ is an optimal solution. Done!
- Else exchange indices between F&G and goto 2



Challenges

1. Computing Large Sums*
 2. Large No. of NNLS problems
- $$x = \begin{bmatrix} C^T \\ C \end{bmatrix}^{-1} \begin{bmatrix} C^T \\ b \end{bmatrix}$$
- Min s.t. $H \geq 0$
- 

* Subscripts from C_F and x_F dropped for simplicity.

Statistical Optimization

The update: $x = (C^T C)^{-1} C^T b$

This is the ordinary least squares(OLS) Estimator: $\min_x \| \begin{bmatrix} C_1^T \\ C_2^T \\ \vdots \\ C_N^T \end{bmatrix} x - \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix} \|_2^2$

Stochastic nature of data – Consider $\{c_i, b_i\}$ as i.i.d instances of RVs $\{c, b\}$.

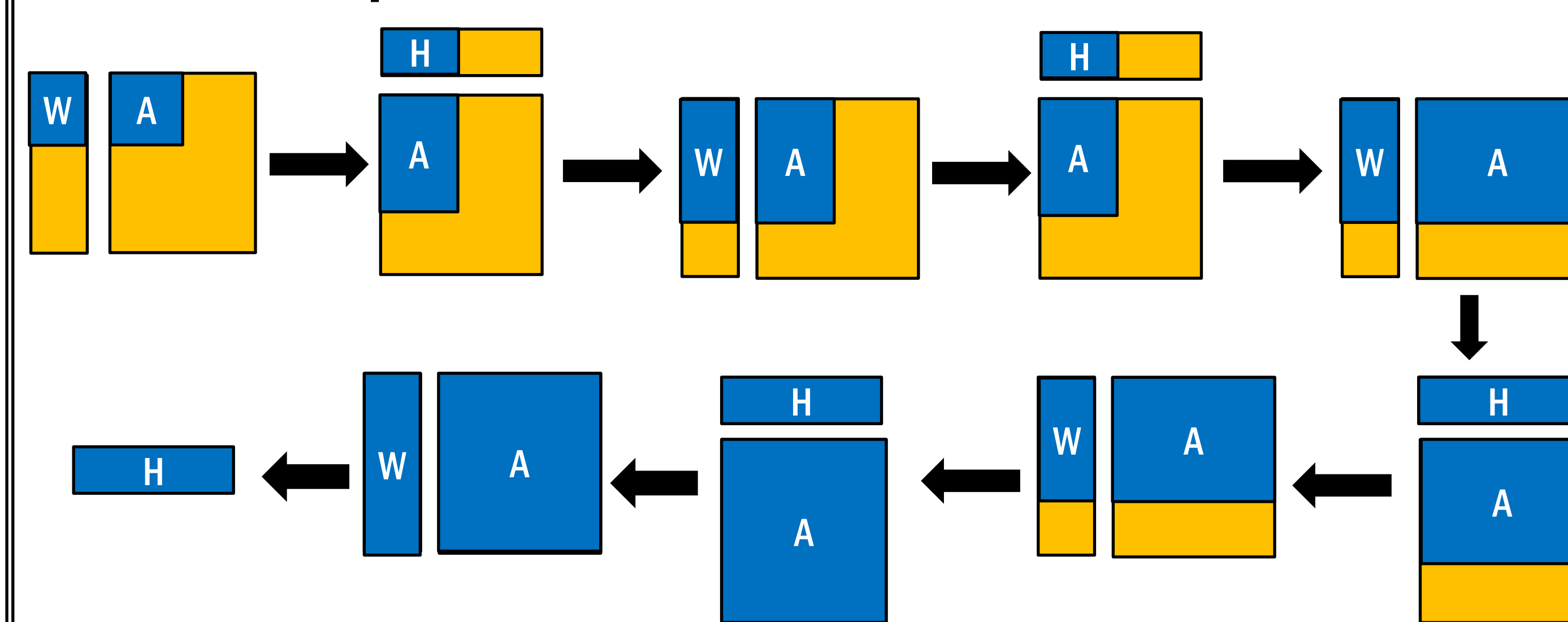
We have the stochastic generative process: $b = C^T x + \epsilon$

Update is asymptotically normal : If we assume

- 1) $Q_{cc} = \mathbb{E}[cc^T]$ is +ve definite
- 2) $\mathbb{E}[\epsilon|c] = 0$
- 3) $Var[\epsilon|c] = \sigma^2$

Then $x \sim \mathcal{N}(\mu, \Sigma)$ where $\Sigma = \frac{Q_{cc} \sigma^2}{N}$. **Note:** variance inversely proportional to N

Statistical Optimization Idea



- Initial iterations / far from optima – we need only a rough update direction. Therefore, use only a few samples of $\{c, b\}$ to save on computation.
- As learning progresses, use more samples to make updates more precise.

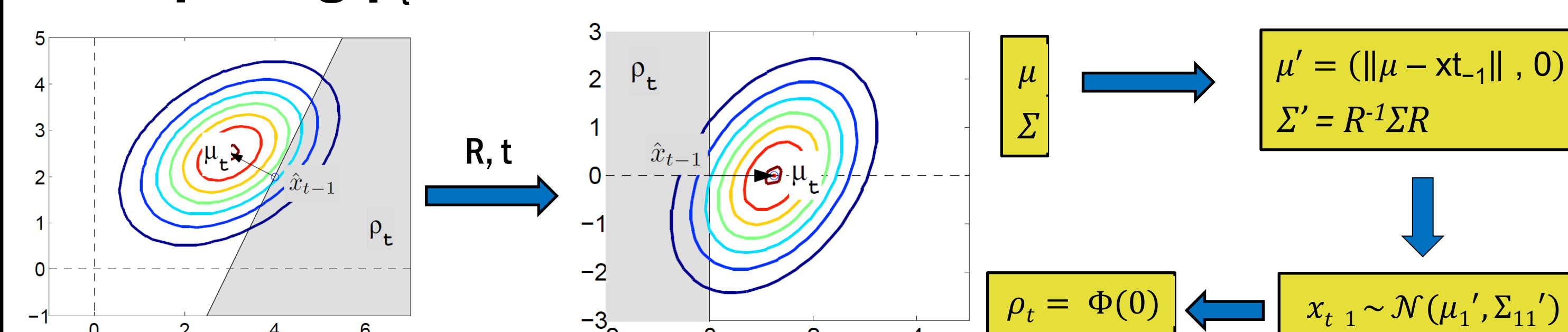
Adaptively determining optimal batch size

- Start with a small batch size.
- Conduct Hypothesis tests to see if update from current batch is reliable.
- If test fails, increase sample size to make updates precise.

Hypothesis Testing

- Test: Proposed update direction is within 90° of the true update direction.
- ρ_t is the probability that our update direction is wrong and should be small.

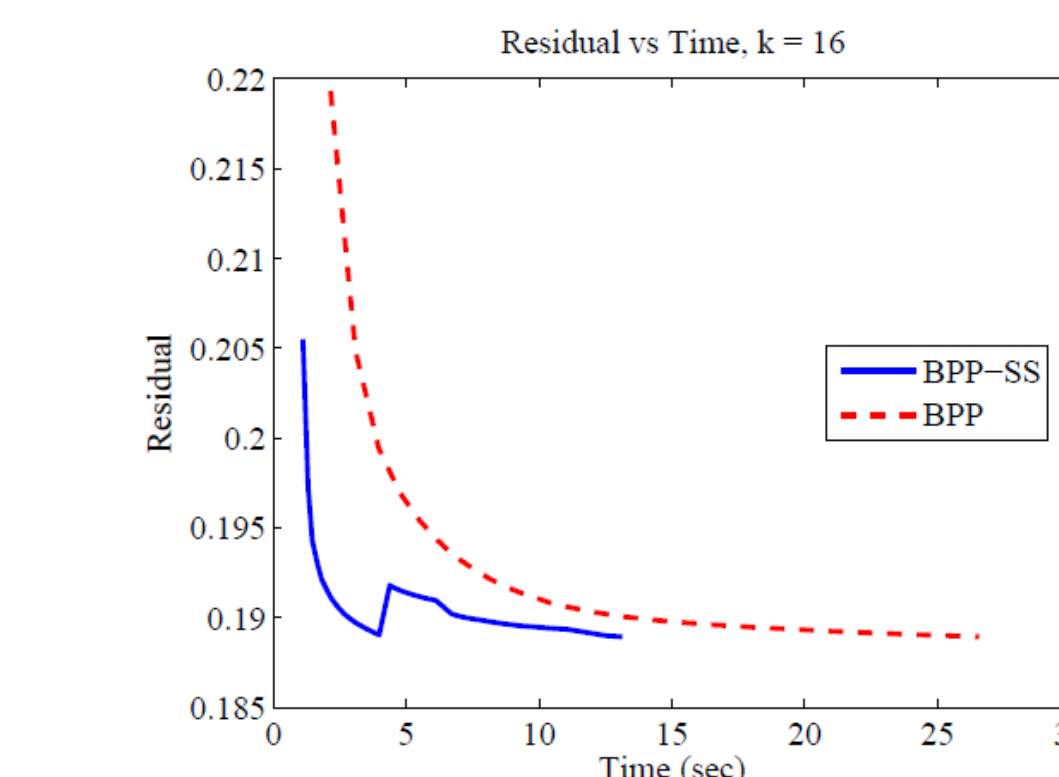
Computing ρ_t :



- If $\rho_t > \Delta$: **Hypothesis Test Fails**. Increase Sample Size.
- Performed only on a random subset of NNLS problems in each step.
- **Principled stopping criterion** – Tests fail when using all available data.

Experimental Results

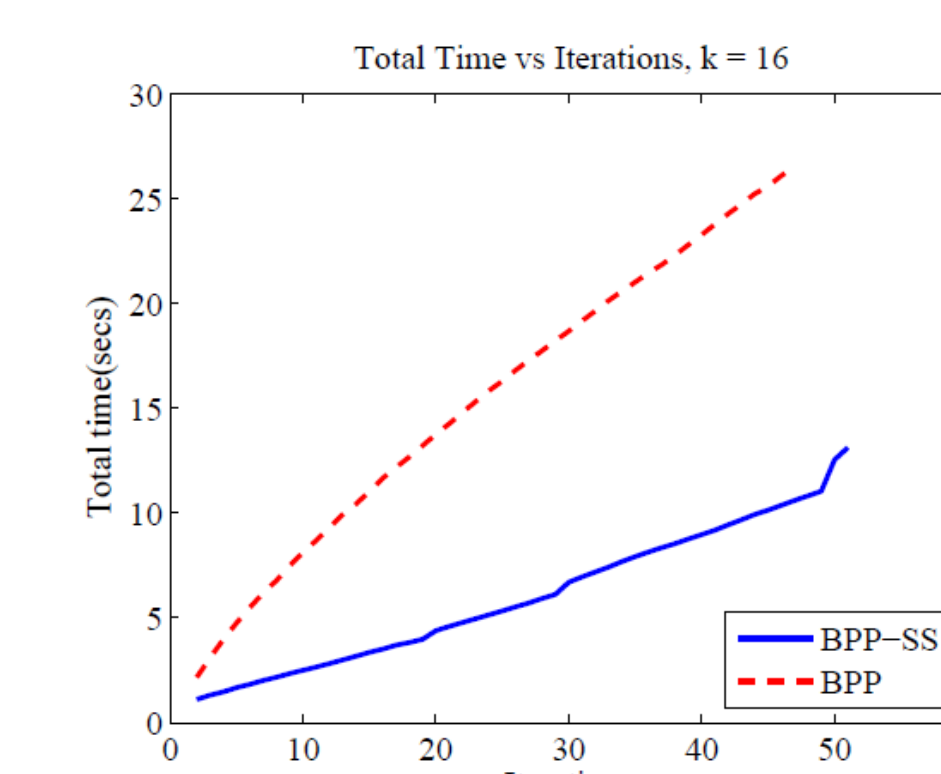
High speed-up over BPP on 3 real world datasets



k	Residual	Running Time (sec)		Speed Up	
		BPP	BPP-SS	Mean	Std. Dev.
16	0.1888	60.53	42.85	1.90	1.21
25	0.1725	106.46	90.57	2.18	1.30
36	0.1591	179.99	143.05	1.95	1.14
49	0.1475	560.64	467.63	2.25	1.26
64	0.1372	407.29	303.99	2.26	1.16
81	0.1284	707.81	459.14	2.28	1.20

AT&T Faces - 10304 x 400 matrix

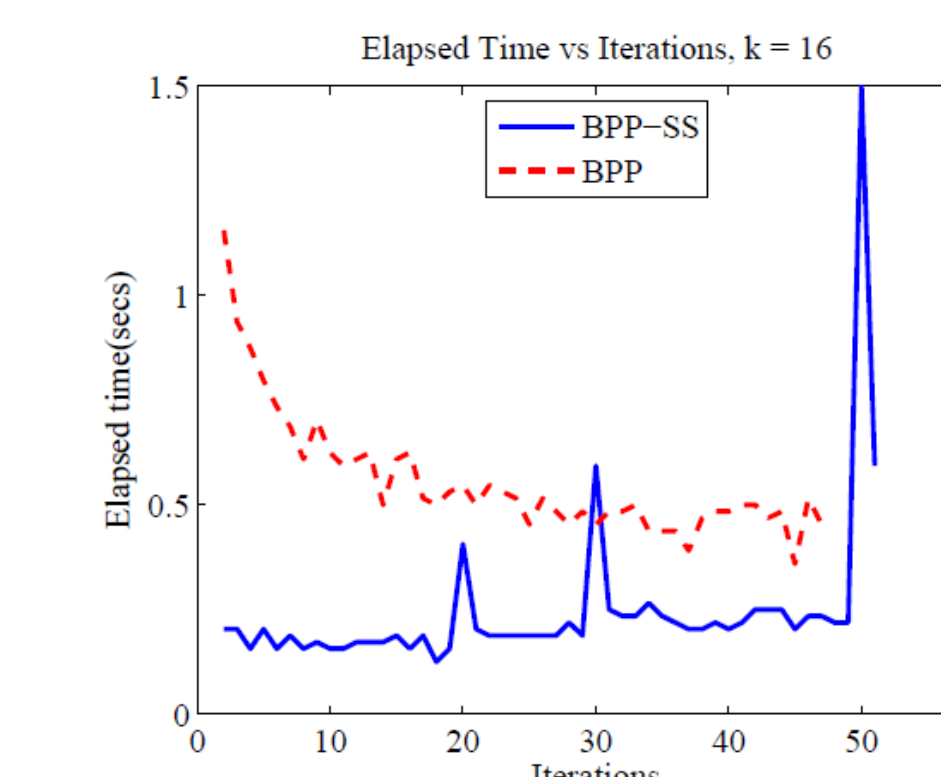
Residual vs Time



Total Time vs Iterations

k	Residual	Running Time (sec)		Speed Up	
		BPP	BPP-SS	Mean	Std. Dev.
10	0.5936	79.2	54.41	1.21	1.43
15	0.5534	271.18	122.39	1.96	1.37
20	0.5228	303.92	150.23	1.96	1.37
25	0.4929	432.79	205.99	2.48	1.50
30	0.4673	578.28	241.86	2.79	1.32
35	0.4441	636.84	262.18	2.44	1.31
40	0.4225	742.26	285.81	2.82	1.30
45	0.4045	1021.1	430.76	2.79	1.54
50	0.3875	1409.9	620.75	2.49	1.41

MNIST - 60000 x 784 matrix

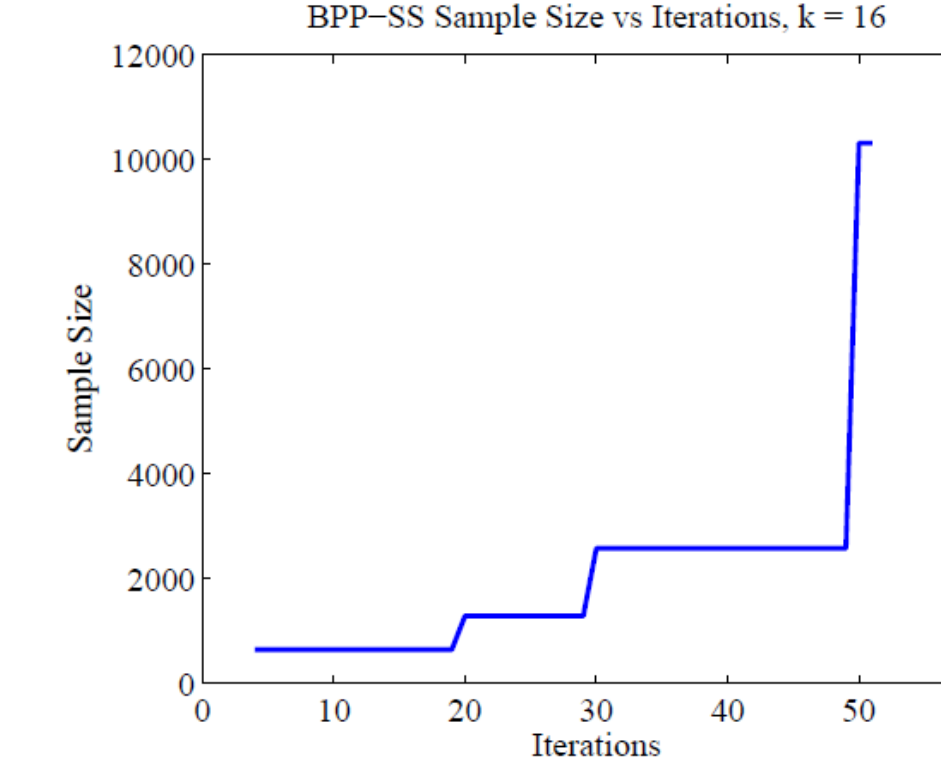


Elapsed Time vs Iterations

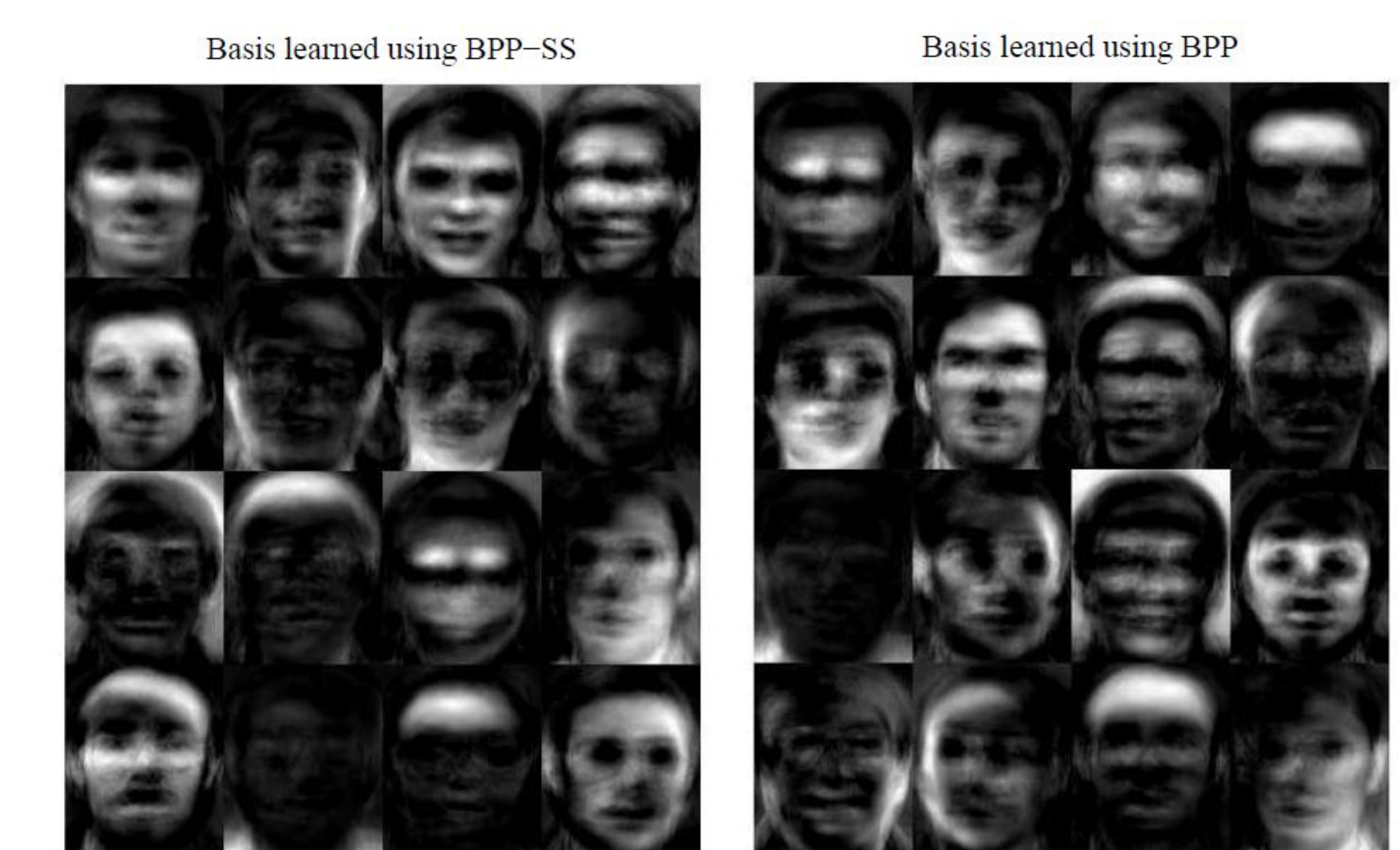
k	Residual	Running Time (sec)		Speed Up	
		BPP	BPP-SS	Mean	Std. Dev.
5	0.3818	2609.9	943.21	2.06	1.79
10	0.3632	7858.4	2084.6	4.20	1.40
15	0.3489	14763	4049.6	3.09	1.46
20	0.3394	13508	5240.6	3.67	1.73
25	0.4894	14976	3929.6	5.39	1.54
30	0.3232	18127	7347.8	5.10	1.38

INRIA HOG - 1000000 x 1092 matrix

Sample Size vs Iterations



Sample Size vs Iterations



AT&T Faces – Comparison of Bases

Conclusions

Advantages:

- High speed-up over BPP
- Single parameter to tune
- Principled stopping criterion, avoids over-fitting
- General method, not limited to NMF

Drawbacks:

Unpredictable if the normality assumptions do not hold e.g. because of sparsity in design matrix, multiplication of random variables etc.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No's. 0447903, 0914783.