libFM & Factorization Machines

Universität Konstanz

Thierry Silbermann

(Most of the) Slides from Steffen Rendle

Application to Large Categorical Domains

User	Movie	Rating
Alice	Titanic	5
Alice	Notting Hill	3
Alice	Star Wars	1
Bob	Star Wars	4
Bob	Star Trek	5
Charlie	Titanic	1
Charlie	Star Wars	5
...	...	...

Feature vector **x** | Target y

	A	B	C	...	TI	NH	SW	ST	...	Target y	
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	5	$y^{(1)}$
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	3	$y^{(2)}$
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	1	$y^{(3)}$
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	4	$y^{(4)}$
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	5	$y^{(5)}$
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	1	$y^{(6)}$
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	5	$y^{(7)}$
	User				Movie						

Applying regression models to this data leads to:

Linear regression: $\hat{y}(\mathbf{x}) = w_0 + w_u + w_i$

Polynomial regression: $\hat{y}(\mathbf{x}) = w_0 + w_u + w_i + w_{u,i}$

Matrix factorization (with biases): $\hat{y}(u,i) = c + w_u + h_i + \langle \mathbf{w}_u, \mathbf{h}_i \rangle$

Factorization Machine (FM)

- Let $\mathbf{x} \in \mathbb{R}^p$ be an input vector with p predictor variables.
- Model equation (degree 2):

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{i=1}^{p} w_i \, x_i + \sum_{i=1}^{p} \sum_{j>i}^{p} \langle \mathbf{v}_i, \mathbf{v}_j \rangle \, x_i \, x_j$$

- Model parameters:

$$w_0 \in \mathbb{R}, \quad \mathbf{w} \in \mathbb{R}^p, \quad \mathbf{V} \in \mathbb{R}^{p \times k}$$

Compared to Polynomial regression:

- Model equation (degree 2):

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{i=1}^{p} w_i x_i + \sum_{i=1}^{p} \sum_{j \geq i}^{p} w_{i,j} \, x_i \, x_j$$

- Model parameters:

$$w_0 \in \mathbb{R}, \quad \mathbf{w} \in \mathbb{R}^p, \quad \mathbf{W} \in \mathbb{R}^{p \times p}$$

Computation Complexity

Factorization Machine model equation:

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{i=1}^{p} w_i \, x_i + \sum_{i=1}^{p} \sum_{j>i}^{p} \langle \mathbf{v}_i, \mathbf{v}_j \rangle \, x_i \, x_j$$

- Trivial computation: $\mathcal{O}(p^2 \, k)$
- Efficient computation can be done in: $\mathcal{O}(p \, k)$
- Making use of many zeros in $\mathbf{x}$ even in: $\mathcal{O}(N_z(\mathbf{x}) \, k)$, where $N_z(\mathbf{x})$ is the number of non-zero elements in vector $\mathbf{x}$.

Efficient Computation

The model equation of an FM can be computed in $\mathcal{O}(p\,k)$.

Proof:

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{i=1}^{p} w_i\, x_i + \sum_{i=1}^{p} \sum_{j>i}^{p} \langle \mathbf{v}_i, \mathbf{v}_j \rangle\, x_i\, x_j$$
$$= w_0 + \sum_{i=1}^{p} w_i\, x_i + \frac{1}{2} \sum_{f=1}^{k} \left[\left(\sum_{i=1}^{p} x_i\, v_{i,f} \right)^2 - \sum_{i=1}^{p} (x_i\, v_{i,f})^2 \right]$$

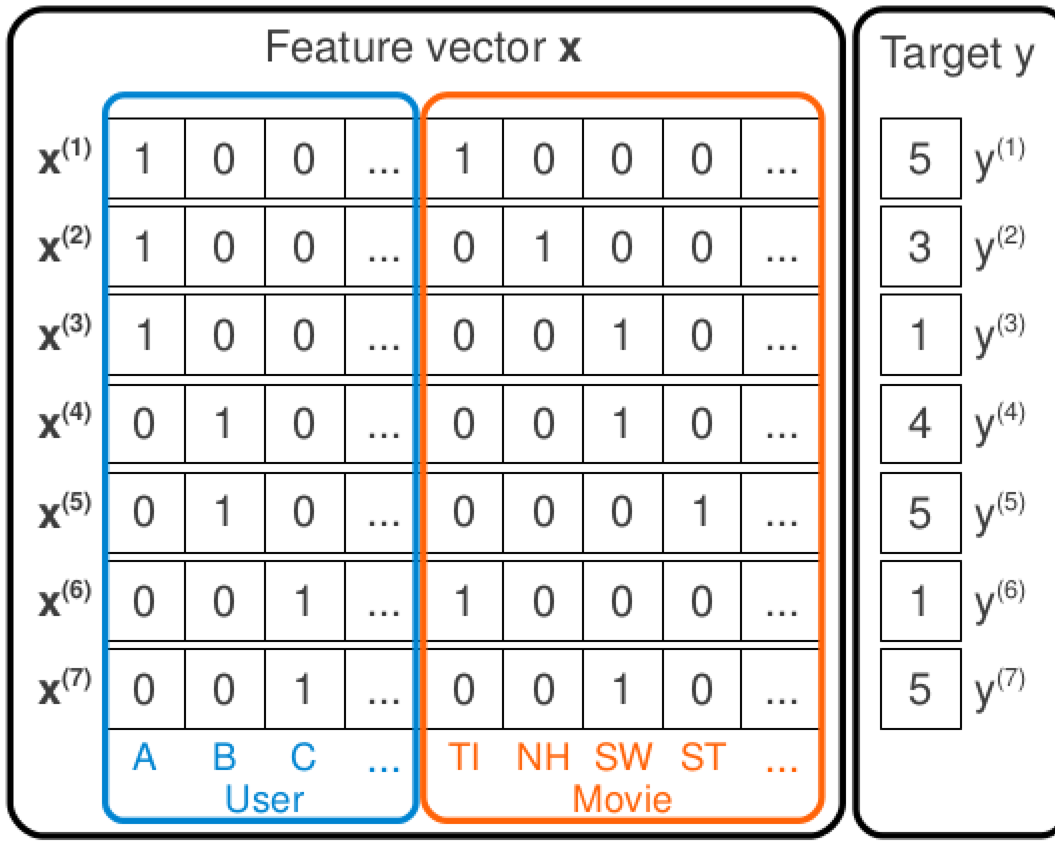
- In the sums over i, only non-zero x_i elements have to be summed up.
- This is the same complexity as the subsumed factorization models (e.g. MF, PITF, Attr-Aware MF, ...).
- (The complexity of polynomial regression is $\mathcal{O}(N_z(\mathbf{x})^2)$.)

Factorization Machines: Discussion

- FMs work with real valued input.
- FMs include variable interactions like polynomial regression.
- Model parameters for interactions are factorized.
- Number of model parameters is $\mathcal{O}(k\, p)$ (instead of $\mathcal{O}(p^2)$ for poly. regr.).

Variable Encoding: Example

User	Movie	Rating
Alice	Titanic	5
Alice	Notting Hill	3
Alice	Star Wars	1
Bob	Star Wars	4
Bob	Star Trek	5
Charlie	Titanic	1
Charlie	Star Wars	5
. . .	. . .	. . .



2 categorical variables

$|U| + |I|$ real valued variables

Matrix Factorization and Factorization Machines

Two categorical variables encoded with real valued predictor variables:

Feature vector $\mathbf{x}$

	User				Movie				
	A	B	C	...	TI	NH	SW	ST	...
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...

With this data, the FM is identical to MF with biases:

$$\hat{y}(\mathbf{x}) = w_0 + w_u + w_i + \underbrace{\langle \mathbf{v}_u, \mathbf{v}_i \rangle}_{\text{MF}}$$

Tag-Recommendation with Factorization Machines

Three categorical variables encoded with real valued predictor variables:

Feature vector **x**

	User: A	User: B	User: C	User: ...	Song: S1	Song: S2	Song: S3	Song: S4	Song: ...	Tag: T1	Tag: T2	Tag: T3	Tag: T4	Tag: ...
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	1	0	0	0	...
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	0	1	0	0	...
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	0	0	0	1	...
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	0	0	1	0	...
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	0	0	1	0	...
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	1	0	0	0	...
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	0	0	0	1	...

With this data, the FM is a tensor factorization model with lower-level interactions (here up to pairwise ones):

$$\hat{y}(\mathbf{x}) := w_0 + w_i + w_u + w_t + \langle \mathbf{v}_u, \mathbf{v}_t \rangle + \langle \mathbf{v}_i, \mathbf{v}_t \rangle + \langle \mathbf{v}_u, \mathbf{v}_i \rangle$$

Attribute-aware MF and Factorization Machines

Two categorical variables and attributes on one of them (here on user) encoded with real valued predictor variables:

Feature vector $\mathbf{x}$

	User				Movie					Other Movies rated				
	A	B	C	...	TI	NH	SW	ST	...	TI	NH	SW	ST	...
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	0.3	0.3	0.3	0	...
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	0.3	0.3	0.3	0	...
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	0.3	0.3	0.3	0	...
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	0	0	0.5	0.5	...
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	0	0	0.5	0.5	...
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	0.5	0	0.5	0	...
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	0.5	0	0.5	0	...

With this data, the FM is identical to:

$$\hat{y}(\mathbf{x}) = \overbrace{w_0 + w_u + w_i + \left\langle \mathbf{v}_i, \mathbf{v}_u + \sum_{l=1}^{\tilde{p}} a_{u,l}\,\mathbf{v}_l \right\rangle}^{\text{Attribute-aware MF}}$$

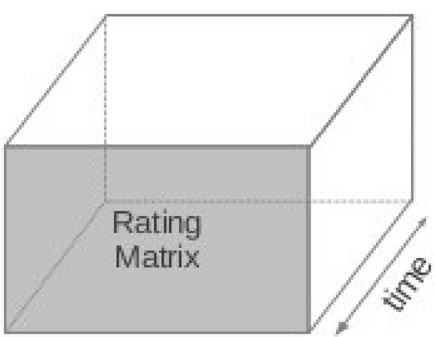
$$+ \sum_{l=1}^{\tilde{p}} \left(a_{u,l}\, w_l + \langle \mathbf{v}_u, a_{u,l}\,\mathbf{v}_l \rangle + \sum_{l'>l} \langle a_{u,l}\,\mathbf{v}_l, a_{u,l'}\;\mathbf{v}'_l \rangle \right)$$

Matrix Factorization & Extensions

Example for data:

Movie

User		TI	NH	SW	ST	...
	A	5	3	1	?	...
	B	?	?	4	5	...
	C	1	?	5	?	...
	...	...	...	...	...	...



Examples for models:

$$\hat{y}^{\mathrm{MF}}(u,i) := \sum_{f=1}^{k} v_{u,f} v_{i,f} = \langle \mathbf{v}_u, \mathbf{v}_i \rangle$$

$$\hat{y}^{\mathrm{SVD++}}(u,i) := \left\langle \mathbf{v}_u + \sum_{j \in N(u)} \mathbf{v}_j, \mathbf{v}_i \right\rangle$$

$$\hat{y}^{\mathrm{Fact\text{-}KNN}}(u,i) := \frac{1}{|R(u)|} \sum_{j \in R(u)} r_{u,j} \langle \mathbf{v}_i, \mathbf{v}_j \rangle$$

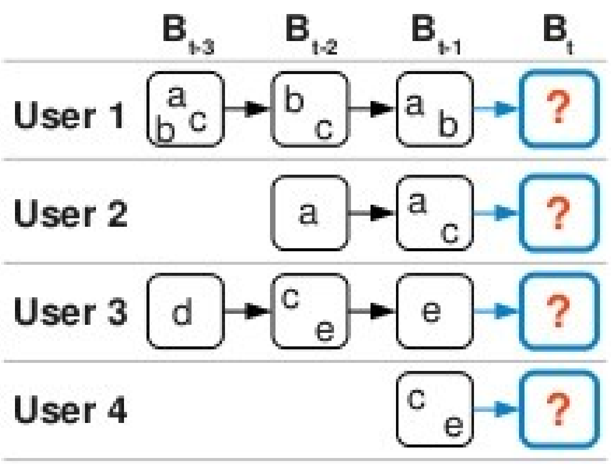
$$\hat{y}^{\mathrm{timeSVD}}(u,i,t) := \langle \mathbf{v}_u + \mathbf{v}_{u,t}, \mathbf{v}_i \rangle$$

$$\hat{y}^{\mathrm{timeTF}}(u,i,t) := \sum_{f=1}^{k} v_{u,f}\, v_{i,f}\, v_{t,f}$$

$$\ldots$$

Sequential Factorization Models

Example for data:



Examples for models:

$$\hat{y}^{\mathrm{FMC}}(u,i,t) := \sum_{l \in B_{t-1}} \langle \mathbf{v}_i, \mathbf{v}_l \rangle$$

$$\hat{y}^{\mathrm{FPMC}}(u,i,t) := \langle \mathbf{v}_u, \mathbf{v}_i \rangle + \sum_{l \in B_{t-1}} \langle \mathbf{v}_i, \mathbf{v}_l \rangle$$

$$\ldots$$

RDF-Triple Prediction with Factorization Machines

Three categorical variables encoded with real valued predictor variables:

Feature vector $\mathbf{x}$

	Subject				Predicate					Object				
	S1	S2	S3	...	P1	P2	P3	P4	...	O1	O2	O3	O4	...
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	1	0	0	0	...
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	0	1	0	0	...
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	0	0	0	1	...
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	0	0	1	0	...
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	0	0	1	0	...
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	1	0	0	0	...
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	0	0	0	1	...

With this data, the FM is equivalent to the PITF model:

$$\hat{y}(\mathbf{x}) := w_0 + w_s + w_p + w_o + \langle \mathbf{v}_s, \mathbf{v}_p \rangle + \langle \mathbf{v}_s, \mathbf{v}_o \rangle + \langle \mathbf{v}_p, \mathbf{v}_o \rangle$$

[PITF: Rendle et al. 2010, WSDM Best Student Paper, ECML 2009 Best DC Award]

Time with Factorization Machines

Two categorical variables and time as linear predictor:

Feature vector $\mathbf{x}$

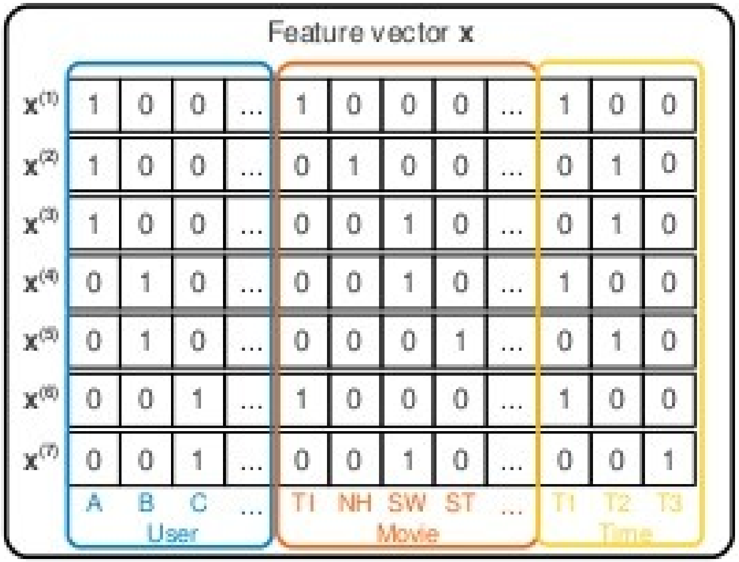
	User				Movie					Time
	A	B	C	...	TI	NH	SW	ST	...	
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	0.2
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	0.6
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	0.61
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	0.3
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	0.5
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	0.1
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	0.8

The FM model would correspond to:

$$\hat{y}(\mathbf{x}) := w_0 + w_i + w_u + t\, w_{\text{time}} + \langle \mathbf{v}_u, \mathbf{v}_i \rangle + t\, \langle \mathbf{v}_u, \mathbf{v}_{\text{time}} \rangle + t\, \langle \mathbf{v}_i, \mathbf{v}_{\text{time}} \rangle$$

Time with Factorization Machines

Two categorical variables and time discretized in bins ($b(t)$):



The FM model would correspond to:[2]

$$\hat{y}(\mathbf{x}) := w_0 + w_i + w_u + w_{b(t)} + \langle \mathbf{v}_u, \mathbf{v}_i \rangle + \langle \mathbf{v}_u, \mathbf{v}_{b(t)} \rangle + \langle \mathbf{v}_i, \mathbf{v}_{b(t)} \rangle$$

---

[2] libFM, $k = 128$, MCMC inference, Netflix RMSE=0.8873

SVD++

Feature vector **x**

	User				Movie					Other Movies rated				
	A	B	C	...	TI	NH	SW	ST	...	TI	NH	SW	ST	...
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	0.3	0.3	0.3	0	...
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	0.3	0.3	0.3	0	...
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	0.3	0.3	0.3	0	...
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	0	0	0.5	0.5	...
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	0	0	0.5	0.5	...
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	0.5	0	0.5	0	...
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	0.5	0	0.5	0	...

With this data, the FM[3] is identical to:

$$\hat{y}(\mathbf{x}) = \overbrace{w_0 + w_u + w_i + \langle \mathbf{v}_u, \mathbf{v}_i \rangle + \frac{1}{\sqrt{|N_u|}} \sum_{l \in N_u} \langle \mathbf{v}_i, \mathbf{v}_l \rangle}^{\text{SVD++}}$$

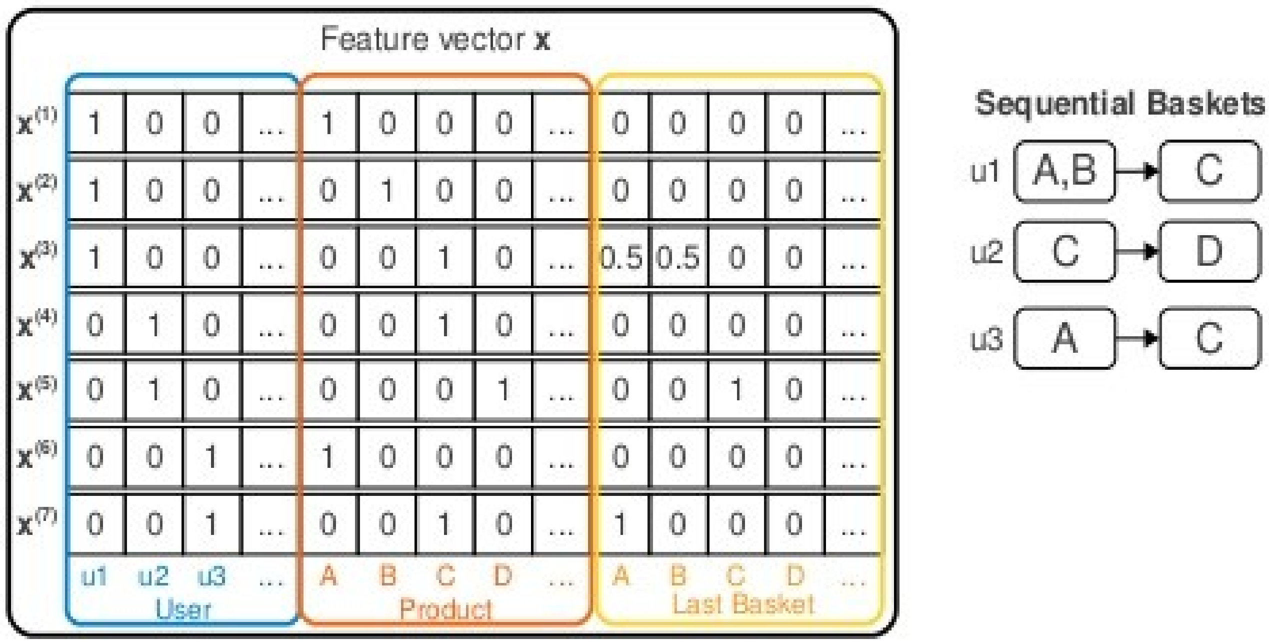
$$+ \frac{1}{\sqrt{|N_u|}} \sum_{l \in N_u} \left(w_l + \langle \mathbf{v}_u, \mathbf{v}_l \rangle + \frac{1}{\sqrt{|N_u|}} \sum_{l' \in N_u, l' > l} \langle \mathbf{v}_l, \mathbf{v}_l' \rangle \right)$$

[Koren, 2008]

[3] libFM, $k = 128$, MCMC inference, Netflix RMSE=0.8865

Factorizing Personalized Markov Chains (FPMC)

Two categorical variables (u,i), one set categorical (B_{t-1}):



FM is equivalent to

$$\hat{y}(\mathbf{x}) := w_0 + w_u + w_i + \frac{1}{|B_{t-1}|} \sum_{j \in B_{t-1}} w_j + \langle \mathbf{v}_u, \mathbf{v}_i \rangle + \frac{1}{|B_{t-1}|} \sum_{j \in B_{t-1}} \langle \mathbf{v}_i, \mathbf{v}_j \rangle + \ldots$$

[Rendle et al. 2010, WWW Best Paper]

(Context-aware) Rating Prediction

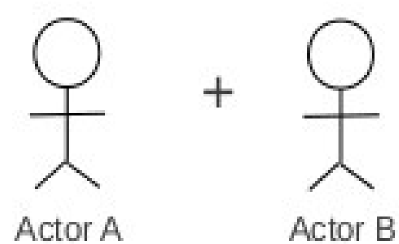
- Main variables:
  - User ID (categorical)
  - Item ID (categorical)
- Additional variables:
  - time
  - mood
  - user profile
  - item meta data
  - ...
- Examples: Netflix prize, Movielens, KDDCup 2011

Netflix Prize

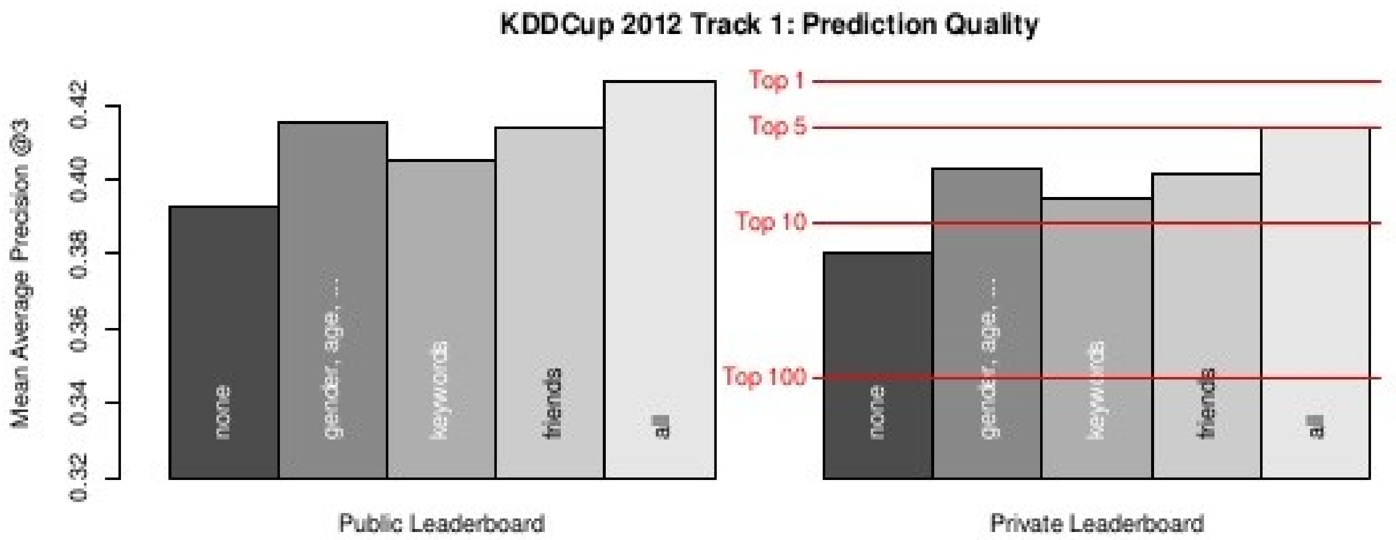
Method (Name)	Ref.	Learning Method	k	Quiz RMSE
Models using user ID and item ID				
Probabilistic Matrix Factorization	[14, 13]	Batch GD	40	*0.9170
Probabilistic Matrix Factorization	[14, 13]	Batch GD	150	0.9211
Matrix Factorization	[6]	Variational Bayes	30	*0.9141
Matchbox	[15]	Variational Bayes	50	*0.9100
ALS-MF	[7]	ALS	100	0.9079
ALS-MF	[7]	ALS	1000	*0.9018
SVD/ MF	[3]	SGD	100	0.9025
SVD/ MF	[3]	SGD	200	*0.9009
Bayesian Probablistic Matrix Factorization (BPMF)	[13]	MCMC	150	0.8965
Bayesian Probablistic Matrix Factorization (BPMF)	[13]	MCMC	300	*0.8954
FM, pred. var: user ID, movie ID	-	MCMC	128	0.8937
Models using implicit feedback				
Probabilistic Matrix Factorization with Constraints	[14]	Batch GD	30	*0.9016
SVD++	[3]	SGD	100	0.8924
SVD++	[3]	SGD	200	*0.8911
BSRM/F	[18]	MCMC	100	0.8926
BSRM/F	[18]	MCMC	400	*0.8874
FM, pred. var: user ID, movie ID, impl.	-	MCMC	128	0.8865

Link Prediction in Social Networks

- Main variables:
  - Actor A ID
  - Actor B ID
- Additional variables:
  - profiles
  - actions
  - ...



KDDCup 2012: Track 1

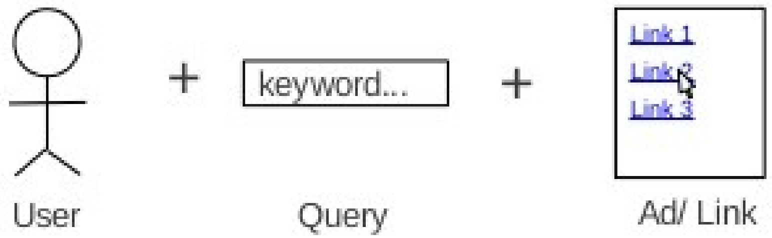


- $k = 22$ factors, 512 MCMC samples (no burnin phase, initialization from random)
- MCMC inference (no hyperparameters (learning rate, regularization) to specify)

[Awarded 2nd place (out of 658 teams)]

Clickthrough Prediction

- Main variables:
  - User ID
  - Query ID
  - Ad/ Link ID
- Additional variables:
  - query tokens
  - user profile
  - ...



KDDCup 2012: Track 2

Model	Inference	wAUC (public)	wAUC (private)
ID-based model ($k = 0$)	SGD	0.78050	0.78086
Attribute-based model ($k = 8$)	MCMC	0.77409	0.77555
Mixed model ($k = 8$)	SGD	0.79011	0.79321
Final ensemble	n/a	0.79857	**0.80178**

Ensemble

- Rank positions (not predicted clickthrough rates) are used.
- The MCMC attribute-based model and different variations of the SGD models are included.

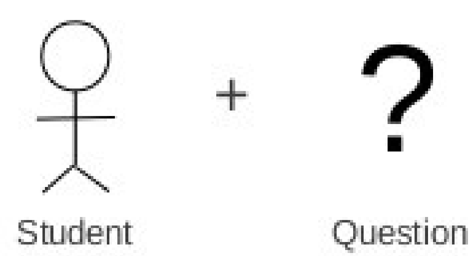
[Awarded 3rd place (out of 171 teams)]

ECML/PKDD Discovery Challenge 2013

- Problem: Recommend given names.
- Main variables:
  - User ID
  - Name ID
- Additional variables:
  - session info
  - string representation for each name
  - ...
- FM approach won 1st place (online track) and 2nd (offline track).

Student Performance Prediction

- Main variables:
  - Student ID
  - Question ID
- Additional variables:
  - question hierarchy
  - sequence of questions
  - skills required
  - ...
- Examples: KDDCup 2010, Grockit Challenge[4] (FM placed 1st/241)



[4] http://www.kaggle.com/c/WhatDoYouKnow

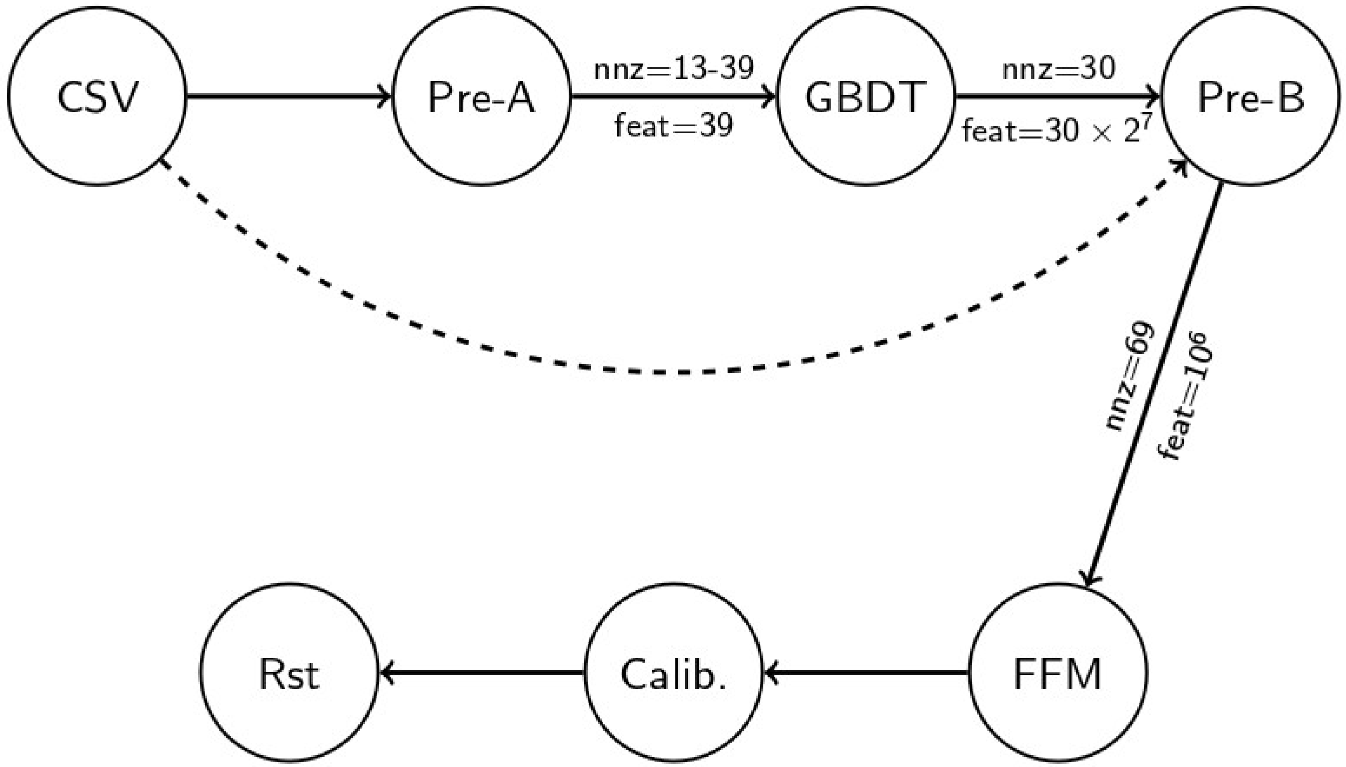
Other Kaggle Competitions

- EMI Music Data Science Hackathon: Best single model is a Factorization Machine with MCMC inference and achieves 13.30247 (private) / 13.27626 (public) [Rendle]

- Blue Book for Bulldozers: Factorization machines [...] gave us our best single model, scoring 0.22450 on the public leaderboard set. We used only the categorical features here [Leustagos, winning team]

- Criteo Display Advertising Challenge: Feature Engineering + Factorization machines

Criteo Display Advertising Challenge



"nnz" means the number of non-zero elements of each impression; "feat" represents the size of feature space.

3 Idiots' Approach for Display Advertising Challenge, Juan et.al
https://github.com/guestwalk/kaggle-2014-criteo

Conclusion

- Representing categorical variables with real-valued variables and applying FMs is comparable to the factorization models that have been derived individually before (e.g. (bias) MF, tensor factorization, attribute-aware MF)
- FMs are much more flexible and can handle non-categorical variables.
- Applying FMs is simple, as only data preprocessing has to be done (defining the real-valued predictor variables)
- Starting to be in the toolbox of every ML people along Random Forest, Vowpal Wabbit, Scikit-learn, Caffe

libFM Software

libFM is an implementation of FMs

- Learning/inference: SGD, ALS, MCMC
- Classification and regression
- Uses the same format as LIBSVM, LIBLINEAR [Lin et. Al], SVMlight [Joachims]
- Support variable grouping
- Open Source: GPLv3

www.libfm.org

https://github.com/srendle/libfm

https://groups.google.com/forum/#!forum/libfm

Thanks

@SilbermannT

thierrysilbermann.wordpress.com (will put some tutorials on libFM)

thierry.silbermann@gmail.com

References

L. Drumond, S. Rendle, and L. Schmidt-Thieme.
Predicting rdf triples in incomplete knowledge bases with tensor factorization.
In *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, SAC '12, pages 326–331, New York, NY, USA, 2012. ACM.

C. Freudenthaler, L. Schmidt-Thieme, and S. Rendle.
Bayesian factorization machines.
In *NIPS workshop on Sparse Representation and Low-rank Approximation*, 2011.

Y. Koren.
Factorization meets the neighborhood: a multifaceted collaborative filtering model.
In *KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434, New York, NY, USA, 2008. ACM.

Y. Koren.
The bellkor solution to the netflix grand prize.
2009.

Y. Koren.
Collaborative filtering with temporal dynamics.
In *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 447–456, New York, NY, USA, 2009. ACM.

Y. J. Lim and Y. W. Teh.
Variational Bayesian approach to movie rating prediction.
In *Proceedings of KDD Cup and Workshop*, 2007.

References

S. Rendle.
Factorization machines.
In *Proceedings of the 2010 IEEE International Conference on Data Mining*, ICDM '10, pages 995–1000, Washington, DC, USA, 2010. IEEE Computer Society.

S. Rendle.
Factorization machines with libFM.
ACM Trans. Intell. Syst. Technol., 3(3):57:1–57:22, May 2012.

S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme.
Factorizing personalized markov chains for next-basket recommendation.
In *WWW '10: Proceedings of the 19th international conference on World wide web*, pages 811–820, New York, NY, USA, 2010. ACM.

S. Rendle, Z. Gantner, C. Freudenthaler, and L. Schmidt-Thieme.
Fast context-aware recommendations with factorization machines.
In *Proceedings of the 34th ACM SIGIR Conference on Reasearch and Development in Information Retrieval*. ACM, 2011.

I. Pilászy, D. Zibriczky, and D. Tikk.
Fast als-based matrix factorization for explicit and implicit feedback datasets.
In *RecSys '10: Proceedings of the fourth ACM conference on Recommender systems*, pages 71–78, New York, NY, USA, 2010. ACM.

I. Porteous, A. Asuncion, and M. Welling.
Bayesian matrix factorization with side information and dirichlet process mixtures.
In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI 2010, pages 563–568, 2010.

References

R. Salakhutdinov and A. Mnih.
Bayesian probabilistic matrix factorization using Markov chain Monte Carlo.
In *Proceedings of the 25th international conference on Machine learning*, ICML '08, pages 880–887, New York, NY, USA, 2008. ACM.

R. Salakhutdinov and A. Mnih.
Probabilistic matrix factorization.
In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1257–1264, Cambridge, MA, 2008. MIT Press.

D. H. Stern, R. Herbrich, and T. Graepel.
Matchbox: large scale online bayesian recommendations.
In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 111–120, New York, NY, USA, 2009. ACM.

L. Xiong, X. Chen, T.-K. Huang, J. Schneider, and J. G. Carbonell.
Temporal collaborative filtering with bayesian probabilistic tensor factorization.
In *Proceedings of the SIAM International Conference on Data Mining*, pages 211–222. SIAM, 2010.

S. Zhu, K. Yu, and Y. Gong.
Stochastic relational models for large-scale dyadic data using MCMC.
In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 1993–2000, 2009.