Learning from Big Data

Mattias Villani

Division of Statistics and Machine Learning
Department of Computer and Information Science
Linköping University

What is Big Data?

- **Volume** - the scale of the data.
  - Financial transactions
  - Supermarket scanners
- **Velocity** - continously streaming data.
  - Stock trades
  - News and social media
- **Variety** - highly varying data structures.
  - Wall street journal articles
  - Network data
- **Veracity** - varying data quality.
  - Tweets
  - Online surveys
- **Volatility** - constantly changing patterns.
  - Trade data
  - Telecom data

Central banks can use big data to ...

- **estimate fine grained economic models more accurately**.
- estimate models for **networks** and flow in network.
- construct **fast economic indicies**:
  - Scanner data for inflation
  - Job adds and text from social media for fine grained unemployment
  - Streaming order data for economic activity
- improve **quality and transparency in decision making**. Summarizing news articles. Visualization.
- improve central banks' **communication**. Is the message getting through? Sentiments. Credibility. Expectations.

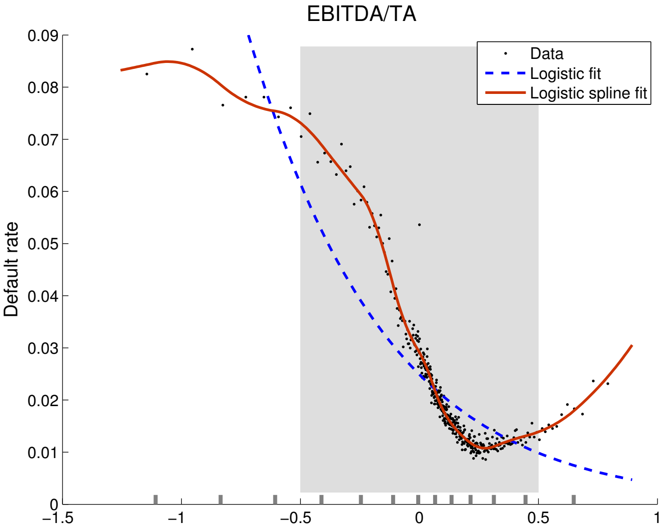
Some recent big data paper in economics

- Varian (2014). *Big data: new tricks for econometrics*. Journal of Economic Perspectives.
- Heston and Sinha (2014). *News versus Sentiment: Comparing Textual Processing Approaches for Predicting Stock Returns*.
- Bholat et al. (2015). *Handbook in text mining for central banks*. Bank of England.
- Bajari et al. (2015). *Machine Learning Methods for Demand Estimation*. AER.

Computationally Big Data

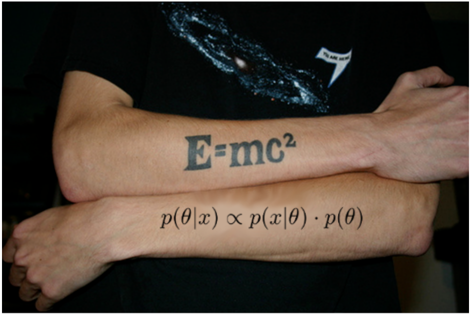
- Data are **computationally big** if they are used in a context where computations are a serious impediment to analysis.
- Even rather small data sets can be computationally demanding when the model is very complex and time-consuming.
- Computational dilemma: model complexity increases with large data:
  - large data have the potential to **reveal poor fit** of simple models
  - with large data one can estimate **more complex** and **more detailed models**.
  - with many observations we can estimate the effect from **more (explanatory) variables**.
- The big question in statistics and machine learning: **how to estimate complex models on large data**?

LARGE DATA REVEALS TOO SIMPLISTIC MODELS



Giordani, Jacobson, Villani and von Schedvin. Journal of Financial and Quantitative Analysis, 2014.

Bayesian learning

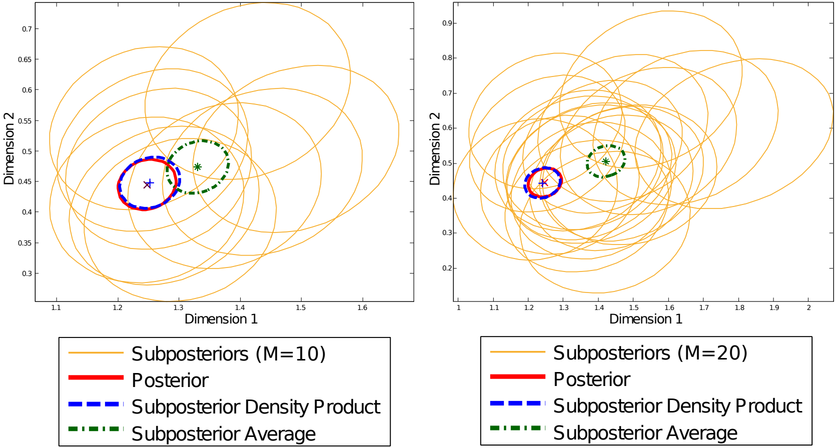


- **Bayesian methods combine** data information with other sources
- ... avoid overfitting by imposing **smoothness** where data are sparse
- ... connect nicely to **prediction** and **decision making**
- ... natural handling of **model uncertainty**
- ... are **beautiful**
- ... are **time-consuming**. MCMC.

Distributed learning for big data

- **Big data** = data that does not fit on a single machine's RAM.
- **Distributed computations**:
  - Matlab: distributed arrays.
  - Python: distarray.
  - R: DistributedR.
- Parallel distributed MCMC algorithms
  - Distribute data across several machines.
  - Learn on each machine separately. MapReduce
  - **Combine the inferences** from each machine in a correct way.

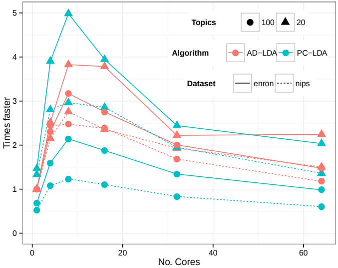
Distributed MCMC



Asymptotically Exact, Embarrassingly Parallel MCMC by Neiswanger, Wang, and Xing, 2014.

Multi-core parallel computing

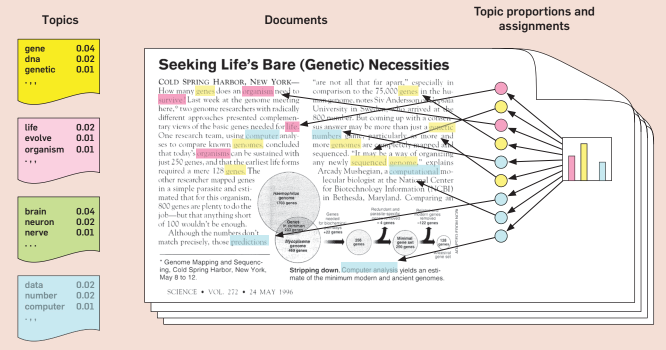
- **Multi-core parallel computing**. Can be combined with distributed computing.
- Available in all high-level languages:
  - Matlab's parallel computing toolbox. `parfor` etc.
  - Python: `multiprocessing` module, `joblib` module etc
  - R: `Parallel` library.
- Communication overheads can easily overwhelm gains from parallelism.



Magnusson, Jonsson, Villani and Broman (2015). Parallelizing LDA using Partially Collapsed Gibbs Sampling.

Topic models

- Probabilistic model for **text**. Popular for summarizing documents.
- Input: a collection of documents.
- Output: K **topics** - probability distributions over the vocabulary. **Topic proportions** for each document.



Blei (2012). Probabilistic Topic Models. Communications of the ACM.

GPU PARALLEL COMPUTING

- **Graphics cards (GPU)** for parallel computing on thousands of cores.
- **Neuroimaging**: brain activity time series in one million 3D pixels.

Table 2 Processing times for three necessary steps in fMRI analysis, for three common software packages, a multicore CPU implementation, and a GPU implementation

Processing step/software	SPM	FSL	AFNI	Multicore CPU	GPU
Motion correction	52 s	36 s	5 s	37 s	1.2 s
Smoothing	31 s	10 s	0.4 s	0.4 s	0.022 s
Model estimation	25 s	4.8 s	0.5 s	0.011 s	0.0008 s

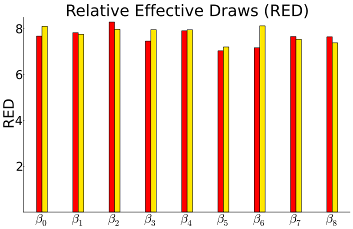
The three common software packages use different algorithms, while the multicore CPU implementation and the GPU implementation perform

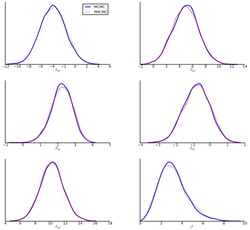
From Eklund, Dufort, Villani and LaConte (2014). Frontiers of Neuroinformatics.

- GPU-enabled functions in
  - Matlab's Parallel Computing Toolbox.
  - PyCUDA in Python.
  - gputools in R.
- Still lots of **nitty-gritty** low level things to get impressive performance: Low-level **CUDA** or **OpenCL** + putting the data in the right place.

Tall data

- **Tall data** = many observations, not many variables.
- Approximate Bayes: VB, EP, ABC, INLA ...
- Recent idea: **efficient random subsampling of the data** in algorithms that eventually give the full data inference.
- Especially useful when likelihood is costly (e.g. optimizing agents).





From Quiroz, Villani and Kohn (2014). Speeding up MCMC by efficient subsampling.

Wide data

- **Wide data** = many variables, comparatively few observation.
- **Variable selection**. Stochastic Search Variable Selection (SSVS).
- **Shrinkage** (ridge regression, lasso, elastic net, horseshoe). Big VARs.

Model Uncertainty in Growth Regressions

Table I. Marginal evidence of importance

		Regressors	BMA Post.Prob.	Sala-i-Martin CDF(0)
$\Rightarrow$	1	GDP level in 1960	1.000	1.000
$\rightarrow$	2	Fraction Confucian	0.995	1.000
$\Rightarrow$	3	Life expectancy	0.946	0.999
$\rightarrow$	4	Equipment investment	0.942	1.000
$\rightarrow$	5	Sub-Saharan dummy	0.757	0.997
$\rightarrow$	6	Fraction Muslim	0.656	1.000
$\rightarrow$	7	Rule of law	0.516	1.000
$\rightarrow$	8	Number of Years open economy	0.502	1.000
$\rightarrow$	9	Degree of Capitalism	0.471	0.987
$\rightarrow$	10	Fraction Protestant	0.461	0.966
$\rightarrow$	11	Fraction GDP in mining	0.441	0.994
$\rightarrow$	12	Non-Equipment Investment	0.431	0.982
$\rightarrow$	13	Latin American dummy	0.190	0.998
$\Rightarrow$	14	Primary School Enrollment, 1960	0.184	0.992
$\rightarrow$	15	Fraction Buddhist	0.167	0.964
	16	Black Market Premium	0.157	0.825
$\rightarrow$	17	Fraction Catholic	0.110	0.963
$\rightarrow$	18	Civil Liberties	0.100	0.997

Wide data

- Many other models in the machine learning literature are of interest: trees, random forest, support vector machines etc.

Table 1—Model Comparison: Prediction Error

	Validation		Out-of-Sample		
	RMSE	Std. Err.	RMSE	Std. Err.	Weight
Linear	1.169	0.022	1.193	0.020	6.62%
Stepwise	0.983	0.012	1.004	0.011	12.13%
Forward Stagewise	0.988	0.013	1.003	0.012	0.00%
Lasso	1.178	0.017	1.222	0.012	0.00%
Random Forest	0.943	0.017	0.965	0.015	65.56%
SVM	1.046	0.024	1.068	0.018	15.69%
Bagging	1.355	0.030	1.321	0.025	0.00%
Logit	1.190	0.020	1.234	0.018	0.00%
Combined	0.924		0.946		100.00%

Bajari et al. (2015). Machine Learning Methods for Demand Estimation. AER.

Online learning

- **Streaming data**. Scanners, internet text, trading of financial assets etc
- How to learn as **data come in sequentially**? Fixed vs time-varying parameters.
- **State space models**:

$$
\begin{aligned}
y_t &= f(x_t) + \epsilon_t \\
x_t &= g(x_{t-1}) + h(z_t) + \nu_t
\end{aligned}
$$

- Dynamic topic models.
- Kalman or particle filters.
- Dynamic variable selection.
- How to **detect changes** in the system online?

Concluding remarks

- Big traditional data (e.g. micro panels) are clearly useful for central banks.
- Remains to be seen if more exotic data (text, networks, internet searches etc) can play an important role in analysis and communication.
- Big data will motivate more complex models. **Big data + complex models = computational challenges**.
- Economists do not have enough competence for dealing with big data. Computer scientists, statisticians, numerical mathematicians will be needed in central banks.
- Economics is not machine learning: not only predictions matter. **How to fuse economic theory and big data**?