we present a notation for describing matrix decomposition models as algebraic expressions, and organize the space of models into a context-free grammar which generates such expressions. The starting symbol corresponds to a structureless model where the entries of the input matrix are modeled as \textit{i.i.d.} Gaussians. Each production rule corresponds to a simple unsupervised learning model, such as clustering or dimensionality reduction. These production rules lie at the heart of our approach: we fit and evaluate a wide variety of models using a small toolbox of algorithms corresponding to the production rules, and the production rules guide our search through the space of structures.

The main contributions of this paper are threefold. First, we present a unifying framework for matrix decompositions based on a context-free grammar which generates a wide variety of structures through the compositional application of a few simple production rules. Second, we exploit our grammar to infer the latent components and estimate predictive likelihood in all of these structures generically and efficiently using a small toolbox of reusable algorithms corresponding to different component matrix priors and production rules. Finally, by performing greedy search over our grammar using predictive likelihood as the criterion, we can (in practice) typically choose the correct structure from the data while evaluating only a small fraction of all possible structures.

Section 3 defines our matrix decomposition formalism, and Sections 4 and 5 present our generic algorithms for inferring component matrices and evaluating individual structures, respectively. Section 6 describes how we search our space of structures. Finally, in Section 7, we evaluate the structure learning procedure on synthetic data and on real-world datasets as diverse as image patches, motion capture, 20 Questions, and Senate voting records, all using \textit{exactly the same code}. Our procedure learns correct and/or plausible model structures for a wide variety of synthetic and real datasets, and gracefully falls back to simpler structures in high-noise conditions.

## 2 Related work

There is a long history of attempts to infer model structures automatically. The field of algorithmic information theory (Li and Vitanyi, 1997) studies how to represent data in terms of a short program/input pair which could have generated it. One prominent example, Solomoff induction, can learn any computable generative model, but is itself uncomputable. Minimum message length (Wallace, 2005), minimum description length (Barron et al., 1998), and Bayesian model comparison (MacKay, 1992) are frameworks which can, in principle, be used to compare very different generative models. In practice, they have primarily been used for controlling complexity within a given model class. By contrast, our aim is to choose from a very large space of model classes by exploiting shared structure between the models.

Other work has focused on searching within more restricted spaces of models, such as undirected (Lee et al., 2006) and directed (Teyssier and Koller, 2005) graphical models, and graph embeddings (Kemp and Tenenbaum, 2008). Kemp and Tenenbaum (2008) model human “domain structure” learning as selecting between a fixed set of graph structures. Similarly to this paper, their structures are generated from a few simple rules; however, whereas their set of structures is small enough to exhaustively evaluate each one, we search over a much larger set of structures in a way that explicitly exploits the recursive nature of the space. Furthermore, our space of matrix decomposition structures is especially broad, including many bread-and-butter models from unsupervised learning, as well as the building blocks of many hierarchical Bayesian models.

We note that several other researchers have proposed unifying frameworks for unsupervised learning which overlap substantially with our own. Roweis and Ghahramani (1999)’s “generative model for generative models” presents a lattice showing relationships between different models. Srebro (2004) and Singh and Gordon (2008) each interpreted a variety of unsupervised learning algorithms as factorizing an input matrix into a product of two factors. Exponential family PCA (Collins et al., 2002; Mohamed et al., 2008) generalizes low-rank factorizations to other observation models in the exponential family. Our work differs from these in that our matrix decomposition formalism is specifically designed to support efficient generic inference and structure learning. We defer discussion of particular matrix decomposition models to Section 3.1, after we have introduced our formalism.

Our work has several parallels in the field of equation discovery. Langley et al. (1984) built a knowledge discovery system called BACON which reproduced classical scientific discoveries. BACON followed a greedy search procedure similar to our own: it repeatedly fit statistical models and looked for additional structure in the learned parameters. Our work is similar in spirit, but uses matrices rather than scalars as the building blocks, allowing us to capture rich structure in high-dimensional spaces. Todorovski and Dzeroski (1997) used a context-free grammar to define spaces of candidate equations. Our approach differs in that we explicitly use the grammar to structure posterior inference and search over model structures.

## 3 A grammar for matrix decompositions

We first present a notation for describing matrix decomposition models as algebraic expressions, such as $MG + G$. Each letter corresponds to a particular distribution over matrices. When the dimensions of all the component matrices are specified, the algebraic expression defines a generative