Abstract

Given a set of relative similarities between objects in the form of triplets “object $i$ is more similar to object $j$ than to object $k$”, we consider the problem of finding an embedding of these objects in a metric space. This problem is generally referred to as triplet embedding. Our main focus in this paper is the case where a subset of triplets are corrupted by noise, such that the order of objects in a triple is reversed. In a crowdsourcing application, for instance, this noise may arise due to varying skill levels or different opinions of the human evaluators. As we show, all existing triplet embedding methods fail to handle even low levels of noise. Inspired by recent advances in robust binary classification and ranking, we introduce a new technique, called $t$-Exponential Triplet Embedding ($t$-ETE), that produces high-quality embeddings even in the presence of significant amount of noise in the triplets. By an extensive set of experiments on both synthetic and real-world datasets, we show that our method outperforms all the other methods, giving rise to new insights on real-world data, which have been impossible to observe using the previous techniques.

Introduction

Learning a metric embedding for a set of objects based on relative similarities is a central problem in human computation and crowdsourcing. The application domain includes a variety of different fields such as recommender systems, ranking, psychology, etc. The relative similarities are usually provided in the form of comparison triplets, where a triplet $(i, j | k)$ expresses that “object $i$ is more similar to object $j$ than to object $k$”, and where the similarity function may be unknown or not even quantified. Object $i$ is usually referred to as the query object and objects $j$ and $k$ are called the test objects. The relative constraints are usually gathered by human evaluators through a data-collecting mechanism such as Amazon Mechanical Turk\footnote{https://www.mturk.com}. These type of constraints have also been used in other applications such as semi-supervised metric learning (Davis et al. 2007; Liu et al. 2012) and clustering (Amid, Gionis, and Ukkonen 2015).

Given a set of relative similarity comparisons on a set of objects, the goal of triplet embedding is to find a representation for the objects in some metric space such that the constraints induced by the triplets are satisfied as much as possible. In other words, the embedding should reflect the underlying similarity measure from which the constraints were generated. Earlier methods for triplet embedding include approaches based on semi-definite programming (Agarwal et al. 2007) and kernel learning (Tamuz et al. 2011). On the other hand, an embedding in which the relative similarity constraints are satisfied only marginally may not necessary provide a useful representation. For that matter, the TSTE method (van der Maaten and Weinberger 2012) adopts heavy-tailed distributions, which results in improved constraint satisfaction and well-separated clusters.

One major drawback of TSTE as well as the other methods is that their performance can drop significantly when a small amount of noise is introduced in the data. The noise may arise due to different reasons. For instance, human evaluators may assume different similarity functions when comparing objects (Amid and Ukkonen 2015). As a result, the order of the test objects may be reversed in a subset of the triplets. One way to handle noisy triplets is, for instance, to map the objects into multiple embeddings where each embedding represents one of the attributes considered for the evaluation (Amid and Ukkonen 2015). However, the problem of mapping the points into a single embedding subject to heavy noise remains largely unsolved.

In this paper, we employ the $t$-exponential distributions and robust loss functions to design a new algorithm for the triplet embedding problem. Our method, $t$-Exponential Triplet Embedding ($t$-ETE)$^2$, inherits the heavy-tail properties of TSTE in producing high-quality embeddings, while being significantly more robust to noise than any other method. Figure 1 illustrates examples of embeddings of a subset of 6000 data points from the MNIST dataset using TSTE (van der Maaten and Weinberger 2012) and our proposed method. The triplets are synthetically generated by sampling a random point from one of the 50-nearest neighbors for each point and another point from those that are located far away (100 triplets for each point). The two embeddings are very similar when there is no noise in the triplets (Figures 1(a) and 1(b)). However, after “reversing” 20% of the triplets, TSTE fails to produce a meaningful embedding (Figure 1(c)) while

---

\(^2\)The acronym TSTE is based on the Student-t distribution. Here, “$t$” is part of the name of the distribution. Our method, $t$-ETE, is based on $t$-exponential family. Here, $t$ is a parameter of the distribution.
t-ETE is almost unaffected by the noise (Figure 1(d)).

The key idea in our approach draws on recent advances in using robust loss functions for binary classification and ranking. Our method eliminates the need for the extra pre-processing step of pruning the triplets and can use all the data collected from a crowdsourcing task. As a result, t-ETE provides new insights about the underlying structure of the data, which was formerly impossible to extract using the previous triplet embedding methods.

### t-Exponential Family of Distributions

We start by introducing the convex, non-negative, non-decreasing function $\exp_t$ with temperature parameter $0 < t < 2$, as a generalization of the standard $\exp$ function (Naudts 2002; 2004b),

$$
\exp_t(x) = \begin{cases} 
\exp(x) & \text{if } t = 1 \\
[1 + (1 - t)x]^1/(1-t) & \text{otherwise}
\end{cases},
$$

where $[\cdot]_+ = \max(0, \cdot)$. The $\exp$ function is recovered in the limit $t \to 1$. Figure 2(a) illustrates the $\exp_t$ function for several values of $t$. Similarly, the $\log_t$ function is defined as the inverse of $\exp_t$,

$$
\log_t(x) = \begin{cases} 
\log(x) & \text{if } t = 1 \\
(x^{1-t} - 1)/(1-t) & \text{otherwise}
\end{cases}.
$$

Note that $\log_t$ is concave and non-decreasing and generalizes the log function which is recovered in the limit $t \to 1$ (see Figure 2(b)). One major difference with the standard $\exp$ and log functions is that the familiar distributive properties do not hold in general: $\exp_t(ab) \neq \exp_t(a) \exp_t(b)$ and $\log_t(ab) \neq \log_t(a) + \log_t(b)$. Note that for values $x > 1$, the $\log_t$ function with $t > 1$ grows slower than the $\log$ function and reaches the value $1/(1-t)$ in the limit $x \to \infty$. This idea can be used to define loss functions which are more robust to noisy constrains, as we will see later.

One important property of $\exp_t$ is that it decays to zero slower than $\exp$ for values of $1 < t < 2$. This motivates defining heavy-tailed distributions using the $\exp_t$ function. More specifically, the $t$-exponential family of distributions is defined as a generalization of the exponential family by using the $\exp_t$ function in place of the standard $\exp$ function (Naudts 2004; Sears 2010). It is easy to verify that the $t$-exponential family includes, as special cases, the Gaussian distribution when $t = 1$, and the Student-t distribution with $\alpha$ degrees of freedom when $-\alpha + 1)/2 = 1/(1-t)$ (see (Ding and Vishwanathan 2010)).

### Triplet Embedding

In this section we formally define the triplet embedding problem. Let $\mathcal{I} = \{1, 2, \ldots, N\}$ denote a set of objects. Suppose that the feature (metric) representation of these objects is unknown. However, some information about the relative similarities of these objects is available in the form of triplets. A triplet $(i, j \mid k)$ is an ordered tuple which represents a constraint on the relative similarities of the objects $i, j,$ and $k$, of the type “object $i$ is more similar to object $j$ than to object $k$.” Let $\mathcal{T} = \{(i, j \mid k)\}$ denote the set of triplets available for the set of objects $\mathcal{I}$.

Given the set of triplets $\mathcal{T}$, the triplet embedding problem amounts to finding a metric representation of the objects,
such that the similarity constraints imposed by the triplets are satisfied as much as possible by a given distance function in the embedding. For instance, in the case of Euclidean distance, we want

\[(i, j \mid k) \implies \|y_i - y_j\| < \|y_i - y_k\|, \text{ w.h.p.} \tag{3}\]

The reason that we may not require all the constraints to be satisfied in the embedding is that there may exist inconsistent and/or conflicting constraints among the set of triplets. This is a very common phenomenon when the triplets are collected through human evaluators via crowdsourcing (Wilber, Kwak, and Belongie 2014; Amid and Ukkonen 2015).

Stochastic Triplet Embedding (STE) (van der Maaten and Weinberger 2012) aims to maximize the joint probability that the triplets \(T\) are satisfied in the embedding \(\mathcal{Y}\). That is, the satisfaction probability of the triplet \((i, j \mid k)\) is defined as

\[p_{ijk} = \frac{\exp(-\|y_i - y_j\|^2)}{\exp(-\|y_i - y_j\|^2) + \exp(-\|y_i - y_k\|^2)}. \tag{4}\]

After forming the probabilities as in Equation (4), STE maximizes the sum of log of the probabilities over the set of triplets \(T\),

\[\max_{\mathcal{Y}} \sum_{(i, j \mid k) \in T} \log p_{ijk} \tag{5}\]

Similarly, TSTE (van der Maaten and Weinberger 2012) replaces the Gaussians in (4) with a Student-t distribution

\[p_{ijk} = \frac{(1 + \frac{\|y_i - y_j\|^2}{\alpha})^{-(\alpha+1)/2}}{(1 + \frac{\|y_i - y_j\|^2}{\alpha})^{-(\alpha+1)/2} + (1 + \frac{\|y_i - y_k\|^2}{\alpha})^{-(\alpha+1)/2}}, \tag{6}\]

and maximizes the sum of log-probabilities over all triplets \(T\), similar to STE. In general, TSTE produces embeddings of higher-quality compared to STE and other triplet embedding methods such as Generalized Non-metric Multidimensional Scaling (GNMDS) (Agarwal et al. 2007) and Crowd Kernel Learning (CKL) (Tamuz et al. 2011).

One major drawback of TSTE as well as STE is their difficulty in handling inconsistent constraints introduced by triplet noise. That is, the performance drops drastically when a small amount of noise contaminates the triplets. The noise could be of different type: contradicting triplets, such as \((i, j \mid k)\) and \((i, k \mid j)\), which cannot be satisfied simultaneously in a single embedding (Amid and Ukkonen 2015), or inconsistent triplets where an attempt to satisfy a particular triplet may cause many other triplets to become unsatisfied. This can happen, for instance, when mapping from a high-dimensional space to a lower dimensional space with fewer degrees of freedom.

Let us consider the objective of STE (and TSTE) in a different format. Using the property \(\log(a) = -\log(1/a)\) of the log function, the maximization problem in (5) can be written in form of the following equivalent minimization problem

\[\min_{\mathcal{Y}} \sum_{(i, j \mid k) \in T} \log \frac{1}{p_{ijk}} \]

\[= \sum_{(i, j \mid k) \in T} \log \left(1 + \frac{\exp(-\|y_i - y_k\|^2)}{\exp(-\|y_i - y_j\|^2)}\right) \]

\[= \sum_{(i, j \mid k) \in T} \log(1 + \varepsilon_{ijk}). \tag{7}\]

The minimization form of the optimization problem offers additional insight about the objective function. The ratio \(\varepsilon_{ijk}\) in (7) can be seen as the non-negative loss of the triplet \((i, j \mid k)\). As the distance \(\|y_i - y_k\|\) increases and the distance \(\|y_i - y_j\|\) becomes small, the loss associated with the triplet approaches zero, i.e., the triplet \((i, j \mid k)\) becomes highly-satisfied. The \(\log(1 + \cdot)\) term on top of each loss term \(\varepsilon_{ijk}\) can be seen as a robust transformation to smooth the effect of each loss. Note that after the transformation, the new loss still grows to infinity as \(\varepsilon_{ijk} \to \infty\), however in a much slower rate. Furthermore, the derivative of the new loss approaches zero in the limit \(\varepsilon_{ijk} \to \infty\). Such loss function are called robust Type-I loss functions (Ding 2013) and have been used in robust binary ranking (Yun, Raman, and Vishwanathan 2014).

The above view can help explain the poor performance of STE and TSTE in the presence of highly-unsatisfied triplets. The loss incurred by such triplets can completely dominate the total loss in (7). The algorithm will try to mitigate for these excess losses by collapsing points together in the center of the map (see Figure 1(c)), such that the pairs of objects in each triplet will tend to become equidistant and hence yield constant loss for each triplet. The key to having a triplet embedding method that can handle high amounts of noise, as in our approach next, is to use a more robust type of loss functions. Namely, we adopt the so-called robust Type-II loss (Ding 2013) which are required to converge to a constant as \(\varepsilon_{ijk} \to \infty\) by using the generalized \(\log_t\) function. Contrary to the Type-I loss used by TSTE, a Type-II loss has a built-in capacity for dealing with excess losses by capping the values to an upper bound, hence offering an automatic mechanism for “giving up” on highly-unsatisfied triplets.

**t-Exponential Triplet Embedding**

The proposed triplet embedding algorithm, t-ETE, leverages the heavy-tailed properties of the t-exponential family and the robustness of Type-II loss functions. Building on our discussion on the heavy-tailed t-exponential families, we define the satisfaction probability of a triplet \((i, j \mid k)\) as

\[p_{ijk}^{(t)} = \frac{\exp_t(-\|y_i - y_j\|^2)}{\exp_t(-\|y_i - y_j\|^2) + \exp_t(-\|y_i - y_k\|^2)}. \tag{8}\]

The numerator in Equation (8) is known as the t-Gaussian distribution (Tanaka 2002) (with zero mean and unit variance) which reduces to the standard Gaussian distribution for \(t = 1\). As we are interested in heavy-tailed distributions, we focus on values of \(1 < t < 2\). Note that for \(t = 1\) and \(t \to 2\), Equation (8) reduces to the STE and TSTE formulation
Figure 3: Generalization and nearest-neighbor performance: MNIST Digits (top row) and MIT Scenes (bottom row). (a) Generalization error, (b) nearest-neighbor error, (c) generalization accuracy in presence of noise, and (d) nearest-neighbor accuracy in presence of noise. For the noise experiments, we use $t = 1.7$ for $t$-ETE.

(with $\alpha = 1$), respectively. In analogy with (7), the objective function in our formulation uses a novel Type-II loss function that involves the $\log_t$ function with $1 < t < 2$:

$$
\min Y \sum_{(i,j,k) \in T} \log_t \frac{1}{p_{ij}(k)} = \sum_{(i,j,k) \in T} \log_t \left( 1 + \exp_t (-\|y_i - y_k\|_2) \right) \exp_t (-\|y_i - y_j\|_2^2). \tag{9}
$$

Note that unlike the standard log function, the $\log_t$ function with $1 < t < 2$ is upper-bounded by the constant value $1/(t-1)$. Additionally, its derivative approaches zero as its argument goes to infinity. Therefore, the $\log_t(1 + \cdot)$ transformation provides a robust Type-II loss for our problem.

We call our method $t$-Exponential Triplet Embedding ($t$-ETE, for short). Note that since $\log_t(a) \neq \log_t(1/a)$ in general, the minimization problem (9) is no longer equal to the maximization problem, defined by summing $\log_t$ of probabilities (8) over $T$. Note that each term in (9) is upper-bounded by $1/(t-1)$. This enables $t$-ETE to handle high levels of noise by simply ignoring those triplets that are highly-unsatisfied. Finally, we note that the probabilities (8) and the robust transformation $\log_t(1 + \cdot)$ in (9) have different roles in the objective. Therefore, we can plausibly generalize the $t$-ETE method by considering different temperature parameters for each quantity, but we do not pursue this idea further in this paper.

Additive vs. Multiplicative Objective

We provide here some additional intuition about our method by analyzing a special case and contrast it to the TSTE method. First, consider the objective function (7) of TSTE by converting the sum of log-probabilities into log of multiplications. Note that we can drop the log function without affecting the global optima of the optimization problem. For $\alpha = 1$, this gives the following equivalent objective function for TSTE:

$$
\max Y \prod_{(i,j,k) \in T} \frac{(1 + \|y_i - y_j\|_2)^{-1}}{(1 + \|y_i - y_j\|_2^{-1} + (1 + \|y_i - y_k\|_2)^{-1} - 1), \tag{10}
$$

which is the product of the satisfaction probabilities (6) over all triplets. Each satisfaction probability term in (10) varies between $[0, 1]$ depending on how well the triplet is satisfied. For highly-unsatisfied triplets, these probabilities become very small. Because of the multiplicative nature of the objective function, even a single highly-unsatisfied triplet can dominate the total product. TSTE tries to mitigate this by collapsing all the points close together, as we argued earlier, resulting in satisfaction probabilities that concentrate around the value $1/2$. 

Figure 4: Embedding of the Food dataset using (a) TSTE, and t-ETE \((t \to 2)\) methods. There appear no clear separation between the clusters in (a) while in (b), three different clusters of food are evident: “Vegetables and Meals” (top), “Ice creams and Deserts” (bottom left), and “Breads and Cookies” (bottom right).

Now, let us consider a special case of the objective function of t-ETE. For \(t \to 2\), it is easy to verify that the objective function in (9) can be written in the following simple form

\[
\max_y \sum_{(i,j) \in T} \frac{(1 + \|y_i - y_j\|^2)^{-1}}{(1 + \|y_i - y_j\|^2)^{-1} + (1 + \|y_i - y_k\|^2)^{-1}}.
\]

(11)

The terms in (11) are the same as the satisfaction probabilities defined for TSTE in (10). However, now the objective involves maximizing the sum of probabilities over all triplets rather than their product. The immediate consequence of having a sum instead of a product is that we can now avoid the highly-unsatisfied triplets by assigning them small probabilities without hurting the total objective. This makes t-ETE robust to high levels of noise in the triplets, contrary to other approaches such as TSTE.

**Experiments**

In this section, we conduct experiments to evaluate the performance of t-ETE compared to the following methods: 1) GNMDS, 2) CKL, 3) STE, and 4) TSTE. We evaluate the generalization performance of the different methods by means of satisfying unseen triplets and the nearest-neighbor error, as well as their robustness to constraint noise. We also provide visualization results on two real-world datasets. The code for the t-ETE method as well as all the experiments will be publicly available soon.

**Datasets**

We consider the following datasets for our experimental evaluation.

**MNIST Digits.** We first randomly subsample 6000 data points from the MNIST handwritten dataset \(^3\) Then, we generate 100 triplets for each data point by considering one point from the 50-nearest neighbors at random and another point from points that are located far away, as the outlier. This yields 600,000 triplets in total.

**MIT Scenes.** We also consider 100 random samples from each class from the MIT Scenes dataset\(^4\) and generate triplets as in the MNIST dataset. This yields 800 objects and 800,000 triplets in total.

**Food Dataset.** The dataset contains 250,320 crowd-sourced triplets queried on 100 different images of food (Wilber, Kwak, and Belongie 2014). We consider all the triplets without any pre-processing to remove inconsistent triplets.

**Music Dataset.** The dataset was gathered via a web-based survey on 1032 users on the similarity of 426 music artists (Ellis et al. 2002). We use the pruned dataset as in (van der Maaten and Weinberger 2012) which contains 9107 triplets on 400 artists. However, the dataset may still contain noisy triplets since the pruning step only discards the inconsistent triplets.

**Generalization and Nearest-Neighbor Error**

We first evaluate the performance of different methods by means of generalization to unseen triplets as well as preserving the nearest-neighbor similarity. For this part of experiments, we consider the MNIST Digits and MIT Scenes datasets. To evaluate the generalization performance, we perform a 10-fold cross validation and report the fraction of held-out triplets that are unsatisfied as a function of number of dimension. This quantity indicates how well the method learns the underlying structure of the data. Additionally, we calculate the nearest-neighbor error as a function of number

\(^3\)http://yann.lecun.com/exdb/mnist/

\(^4\)http://people.csail.mit.edu/torralba/code/spatialenvelope/
of dimensions. The nearest-neighbor error is a measure of how well the embedding captures the pairwise similarity of the objects based on relative comparisons. The results are shown in Figure 3(a)-3(b). As can be seen, t-ETE performs as good as the best performing method or even better on both generalization and nearest-neighbor error. This indicates that t-ETE successfully captures the underlying structure of the data and scales properly with the number of dimensions.

Robustness to Noise

Next, we evaluate the robustness of the different methods to triplet noise. To evaluate the performance, we generate a different test set for both datasets with the same number of triplets as the training set. For each noise level, we randomly subsample a subset of training triplets and reverse the order of the objects. After generating the embedding, we evaluate the performance on the test set and report the fraction of the test triplets that are satisfied as well as the nearest-neighbor accuracy. The results are shown in Figure 3(c)-3(d). As can be seen, the performance of all the other methods starts to drop immediately when only a small amount of noise is added to the data. On the other hand, t-ETE is very robust to triplet noise such that the performance is almost unaffected for up to 15% of noise. This verifies that t-ETE can be effectively applied to real-world datasets where a large portion of the triplets may have been corrupted by noise.

Visualization Results

We provide visualization results on the Food and Music datasets. Figures 4(a) and 4(b) illustrate the results on the Food dataset using TSTE and t-ETE (t → 2), respectively. The same initialization for the data points is used for the both methods. As can be seen, no clear clusters are evident using the TSTE method. On the other hand, t-ETE reveals three main clusters in the data: “Vegetables and Meals” (top), “Ice creams and Deserts” (bottom left), and “Breads and Cookies” (bottom right).

We also provide a visualization of the Music dataset using the t-ETE method (t → 2). The result can be compared with the one using the TSTE method, available on http://homepage.tudelft.nl/19j49/stc. The distribution of the artists and the neighborhood structure are similar for both methods, but more meaningful in some regions using the t-ETE method. This can be due to the noise in the triplets that have been collected via human evaluators. Additionally, t-ETE results in 0.52 nearest-neighbor error on the data points compared to 0.63 error using TSTE.

Conclusion

We introduced a new triplet embedding method, t-ETE, which is highly robust to noise in the constraints caused by human evaluators. Our method is based on the heavy-tailed properties of t-exponential family of distributions and robust loss functions used in binary ranking and classification. t-ETE produces high-quality results, even in presence of significant amount of noise. This can be critical in cases where the triplets are gathered via human evaluators and the noise in the comparisons is almost unavoidable due to different skill levels or opinions of the evaluators. As a result, t-ETE successfully unveils the underlying structure of the real-world datasets which have not been possible to observe using the previous triplet embedding techniques.
References

Agarwal, S.; Wills, J.; Cayton, L.; Lanckriet, G.; Kriegman, D.; and Belongie, S. 2007. Generalized non-metric multidimensional scaling. In Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics.

Amid, E., and Ukkonen, A. 2015. Multiview triplet embedding: Learning attributes in multiple maps. In Proceedings of the 32nd International Conference on Machine Learning (ICML-15), 1472–1480.

Amid, E.; Gionis, A.; and Ukkonen, A. 2015. A kernel-learning approach to semi-supervised clustering with relative distance comparisons. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 219–234. Springer.

Davis, J. V.; Kulis, B.; Jain, P.; Sra, S.; and Dhillon, I. S. 2007. Information-theoretic metric learning. In Proceedings of the 24th international conference on Machine learning, 209–216. ACM.

Ding, N., and Vishwanathan, S. V. N. 2010. $t$-logistic regression. In Proceedings of the 23th International Conference on Neural Information Processing Systems, NIPS’10, 514–522.

Ding, N. 2013. Statistical machine learning in the $t$-exponential family of distributions. Ph.D. Dissertation, Purdue University.

Ellis, D. P. W.; Whitman, B.; Berenzweig, A.; and Lawrence, S. 2002. The Quest for Ground Truth in Musical Artist Similarity. In Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR ’02), 170–177.

Liu, E. Y.; Guo, Z.; Zhang, X.; Jojic, V.; and Wang, W. 2012. Metric learning from relative comparisons by minimizing squared residual. In 2012 IEEE 12th International Conference on Data Mining, 978–983. IEEE.

Naudts, J. 2002. Deformed exponentials and logarithms in generalized thermostatistics. Physica A 316:323–334.

Naudts, J. 2004a. Estimators, escort probabilities, and phi-exponential families in statistical physics. Journal of Inequalities in Pure and Applied Mathematics 5(4):102.

Naudts, J. 2004b. Generalized thermostatistics based on deformed exponential and logarithmic functions. Physica A 340:32–40.

Sears, T. 2010. Generalized Maximum Entropy, Convexity and Machine Learning. Ph.D. Dissertation, The Australian National University.

Tamuz, O.; Liu, C.; Belongie, S.; Shamir, O.; and Kalai, A. T. 2011. Adaptively Learning the Crowd Kernel. In Proceedings of the 28th International Conference on Machine Learning (ICML-11).

Tanaka, M. 2002. A consideration on a family of $q$-normal distributions. IEICE transactions on information and systems 85(2):435–436.

van der Maaten, L., and Weinberger, K. 2012. Stochastic triplet embedding. In 2012 IEEE International Workshop on Machine Learning for Signal Processing, 1–6.

Wilber, M.; Kwak, S.; and Belongie, S. 2014. Cost-effective hits for relative similarity comparisons. In Human Computation and Crowdsourcing (HCOMP).

Yun, H.; Raman, P.; and Vishwanathan, S. V. N. 2014. Ranking via robust binary classification. In Proceedings of the 27th International Conference on Neural Information Processing Systems, NIPS’14, 2582–2590.