CSE 255 Assignment 9

Alexander Asplund, William Fedus

September 25, 2015

1 Introduction

In this paper we train a logistic regression function for two forms of link prediction among a set of 244 suspected terrorists in a social network. We train and test on a dataset created at the University of Maryland and further modified at UCSD by Eric Doi and Ke Tang [2]. The supposed terrorists have several labels for the nature of their links to other supposed terrorists; terrorists are classified as either colleagues, family, contacts, or congregates. Structural information about the known network connectivity of the supposed terrorists is integrated with additional binary information provided about the individuals to arrive at two final models. The first model predicts the existence of any type of link between two individuals and the second model classifies whether an existing link is 'colleague' or 'other'. In the link prediction task, our final logistic regression, with per-example cost of 117, generates an average AUC metric 0.93 and on the second link classification task, the final linear logistic regression, with per-example regularization of 33.7, generates a 0.92 0/1 accuracy metric.

2 Data Statistics

The Terrorists dataset contains 612 binary features of the 244 supposed terrorists and also structural information about the network with 840 agent-agent known links. The links between agents have four possible labels, with the following base rates:

- *Colleague*: 487 (53.1%)
- *Family*: 136 (14.8%)
- *Contact*: 114 (19.6%)
- *Congregate*: 180 (12.4%)

To extract the structural information of the network, we first construct a symmetric 244 $\times$ 244 adjacency matrix, A, where 0 represents an unknown or unspecified link between agents and 1 represents a known positive link. In

this paper, we assume that all links are transitive and thus undirected, so any directed links are explicitly imputed so that if agent i is linked with agent j, agent j is linked with agent i.

The Terrorists dataset also contains 612 binary descriptive features for each agent in the pair, totaling 1224 binary attributes, however, the exact interpretation of the the attributes is not available.

3 Training Methodology

We use a linear logistic regression model in order to incorporate the descriptive and the structural features of the dataset. For both classification tasks, 10-fold stratified cross-validation is performed in both tasks to select the regularization parameter C which optimizes the AUC metric of both link prediction tasks. In both tasks, the objective is to determine the weight vector $\mathbf{w}$ which minimizes the training loss

$$\mathbf{w} = \text{argmin}_{\mathbf{w}\in\mathbb{R}^d} \|\mathbf{w}\|^2 + C_{pos} \sum_{i=1}^{n} \text{loss}(f(\mathbf{x};\mathbf{w}), y_i) + C_{neg} \sum_{i=1}^{n} \text{loss}(f(\mathbf{x}), y_i) \quad (1)$$

where we use a logistic loss function, C_{pos} and C_{neg} represent the regularization parameters for positive and negative examples respectively, y_i represents the label for training example i and $f(\mathbf{x};\mathbf{w})$ is the sigmoid function

$$f(\mathbf{x};\mathbf{w}) = \frac{1}{1+e^{-\mathbf{w}\cdot\mathbf{x}}} \quad (2)$$

Additionally, the two regularization parameters C_{pos} and C_{neg} are set simply according to the ratio of positive (N_{pos}) and negative training examples (N_{neg}) in a given training fold; therefore, this model reduces to one with only a single regularization parameter C with positive examples scaled by N_{neg}/N_{pos}.

3.1 Descriptive Features

The descriptive binary features of the agents are first used to create training sets which depend on information not included in the network structure. To predict a link between terrorist i and terrorist j, we first define the vector of features for each to be **da** and **db**, respectively. We choose the elementwise product of **da** and **db** as a transformed feature set to allow for the interaction of the features between supposed terrorist i and j. We will refer to the transformation as *D-pt*, written below

- *D-pt*: $< da_1db_1, da_2db_2, ..., da_{612}db_{612} >$

3.2 Structural Network Features

Rank-k Singular Value Decomposition (SVD) is used to extract the structural network information of the agent-agent connections. The full rank SVD of matrix A, as written in Equation 3

$$A = USV^T \tag{3}$$

may be approximated via the top k singular values of the original S matrix and rewritten as

$$A = A' \approx US'V^T \tag{4}$$

where S' only takes the first k singular values from S. In order to predict links within the network during 10-fold cross validation, we partition the 244 $\times$ 243 /2 possible links into ten subsets. During each cross-validation iteration, those links in the cross-validation paritition are declared as unknown in the adjacency matrix and then imputed according to column averages. Computing the SVD on this new matrix and then using only the first k singular values serves as a prediction matrix for the original matrix A.

Each fold of the imputed A' matrix may be written as

$$A' \approx US'V^T = U\sqrt{S'}\sqrt{S'}V^T \coloneqq U_S V_S^T \tag{5}$$

where the S' matrix is split into both U and V matrices. From this representation, features of the structure of the network may be extracted; a feature vector **sa** will be the i^{th} column of U_S and **sb** will be the j^{th} row of V_S^T. Two additional feature sets may be constructed from these two vectors

- *SVD-dot*: $\mathbf{sa} \cdot \mathbf{sb} = A_{ij}$
- *SVD-pt*: $< sa_1 sb_1, sa_2 sb_2, ..., sa_{612} sb_{612} >$

These two feature sets may now be used for the prediction of links between agents in the logistic regression.

3.3 Design of Experiments

We first run a series of experiments to determine which combinations of the derived descriptive and structural feature sets to use in the final models for both tasks of link prediction and link classification. All subsets of the original three feature sets are tested, except for the two with the combination of *SVD-dot* *SVD-pt* (*SVD-pt* is simply a compressed information element derived from *SVD-pt* and is therefore redundant to include in any experiment with *SVD-pt*) and *SVD-dot* individually which contains less information than *SVD-pt* individually. In order to avoid the feature representation failure of [1], we do not choose a descriptive feature set which is simply the concatenation of vectors **da** and **db**. This concatenated feature set does not allow a linear classifier to learn

patterns true for specific agents and only learns the propensity of the agents to be connected on a global basis.

Also, during the extraction of structural information, the SVD of the original matrix must not include the connectivity information of the test partition. Therefore, when performing cross-validation, it is crucial that connectivity information of the agents in the test set must be set to unknown and then imputed before the feature sets are created from matrices U_S and V_S^T, otherwise, there will be a leakage of information about the future network structure into the learning model. It is not correct to find the SVD of the entire matrix and then perform cross-validation, the adjacency matrix must be recomputed for each fold of cross-validation.

The issue of dealing with the structural network information is further complicated by the fact that there is considerable ambguitity whether a 0 entry in matrix A represents a negative indication of a link between agents or is simply a missing value. We believe that the rank-k SVD begins to address this issue by transforming this sparse adjacency matrix A into A' which has a set of scores for each link, indicating a likelihood of that link existing. This provides the logisitic regression model with a set of training features which discriminate between negative values and missingness through an estimation procedure of rank-k SVD.

Additionally, as a basic assessment of the statistical significance of our design experiments, we also include the standard deviation of the AUC metric, derived from the 10-folds of the training procedure. This allows us to more clearly assess the differences in performance between our feature sets and have confidence the results are not statistical aberrations.

Finally, once the optimal feature set is determined, as an additional data transformation, we test the use of a standard Laplacian Transform on the original adjacency matrix A. Here we use the standard Laplancian Matrix definition of $L = D - A$ where D is the degree matrix and each l_{ij} in L is equal to $deg(a_i)$ if $i = j$ or -1 if $i \neq j$.

3.4 Results of Experiments

We use 10-fold stratified cross-validation and perform training on combinations of feature sets, comparing which result in the highest average AUC metric over cross-validation folds.

From Table 1 we can see that the AUC for link prediction is maximized with the $\{SVD\text{-}dot\ D\text{-}pt\}$ feature sets and from Table 2, that the AUC for link *classification* is maxmized using the $\{SVD\text{-}pt\}$ feature set.

4 Results

Our series of experiments indicated that for the link prediction task, the AUC metric would be maximized with a logisitic regression using feature sets $\{SVD\text{-}dot\ D\text{-}pt\}$. There is, however, a latent risk of leakage of information from D_pt since these

Link Prediction

Feature Sets	AUC Metric	Std Dev
SVD-dot D-pt	0.87	± 0.03
SVD-pt D-pt	0.83	± 0.01
SVD-pt	0.83	± 0.02
Laplace SVD-dot D-pt	0.79	± 0.03
D-pt	0.65	± 0.03

Table 1: Experimental results of using different combinations of descriptive and structural feature sets for link prediction.

Link Classification

Feature Sets	AUC Metric	Std Dev
SVD-pt	0.93	± 0.03
Laplace SVD-pt	0.92	± 0.02
SVD-pt D-pt	0.91	± 0.03
SVD-dot D-pt	0.78	± 0.04
D-pt	0.66	± 0.03

Table 2: Experimental results of using different combinations of descriptive and structural feature sets for link classification.

[b]0.7

Figure 1: AUC scores for link type classification task

[b]0.7

Figure 2: AUC scores for link existence prediction task

Figure 3: Experimental results

node binary variables are all unlabeled and thus it is impossible to know whether these labels may contain information about the classification goal.

The link prediction final model achieves an average AUC metric of 0.93 on our 10 cross-validation folds. For link classification, the AUC metric is found to be maxmized using features $SVD - pt$ and the final link classification model generates an average AUC metric of 0.87 and an average 0/1 accuracy metric of 0.92 on our 10 cross-validation folds. Since the Colleague class is quite balanced with a base rate of 53.1% the 0/1 accuracy can be considered a good measure of success.

In Table 3 we present the confusion matrices for both final models. We derive these by taking the mean of true positives, true negatives, false positives and false negatives across the 10 cross validation iterations and then normalizing it.

For the final link prediction model, the recall is 0.79 and the precision is 0.42; for the final link classification model,the recall is 0.86 and the precision is 0.99.

		Actual	
		Neg	Pos
Predict	Neg	94.2%	0.6%
	Pos	3.0%	2.2%

		Actual	
		Neg	Pos
Predict	Neg	45.1%	7.8%
	Pos	0.7%	46.4%

Table 3: Final confusion matrices for both link prediction (left) and link classification (right).

5 Conclusions

Our results are largely inline with those reported in [2], which uses the same dataset. We observe that *SVD-pt* and/or *SVD-ptD-pt* achieve the highest average AUC metric on link prediction, while *SVD-dot D-pt* achieves the highest AUC metric on link type classification. This result that *SVD-pt* is better for existence prediction while *SVD-dot* is better for type classification is also seen in [2], though for a different dataset.

The best link classification model uses the $SVD-ptD$-*pt* feature set. However, it is not known specifically what these features represent and we warn that this might be a source of leakage of information during cross-validation. Since the AUC metric with *SVD-pt* alone is 0.82 versus the 0.87 of *SVD-ptD-pt*, it is worth considering selecting the former model to support higher confidence in future performance.

References

[1] Joel R. Bock and David A. Gough. Predicting protein-protein interactions from primary structure. *Bioinformatics*, pages 455–460, 2001.

[2] Eric Doi. Low-rank decomposition and logistic regression methods for link prediction in terrorist networks. 2010.