COS 424 Homework 3. Latent Feature Analysis of OkCupid Profiles : The Relationship Between Number of Latent Groups and “Tweener” Users

Francisco J. Carrillo
Department of Chemical and Biological Engineering
fjc2@princeton.edu

Abstract

The study and identification of the hidden (i.e. latent) features on data sets has far-reaching implications in the fields of data science. Potential (and current) applications of latent feature analysis includes the development of search engines, the creation of stock trading algorithms, population analysis, and the sorting of people into groups (for commercial, dating, and/or policy purposes). In this report we use “One-Hot” encoding, natural language processing (“Bag of Words”), and Latent Dirichlet Allocation to process and analyze the data from 59946 real OkCupid dating profiles originating from the San Francisco Bay Area. In particular we study the relationship between the prevalence of “Tweeners” (users who are sorted into several groups) and the hyper parameter “k” (total number of groups). We conclude that, for this data set, “k” has an optimal value of 7, which dramatically decreases the number of tweeners while still being an intepretable and manageable number of groups. The resulting analysis sorted users into groups composed of 1) intellectuals 2) educated white people 3) artistic people, 4) active people, 5) hipsters, 6) people who love life, and 7) social people. Furthermore, we identified that tweeners tend to be part of a group which consists of ”drug-loving atheists”.

1 Introduction

Latent feature analysis and its ability to identify hidden features in large data sets has lead to the development of services that are used daily by millions of people: search engines (Google), dating websites (OkCupid), video services (Netflix), and music streaming services (Spotify). There are many techniques used to identify underlying hidden (latent) trends in data. Some of them specialize on obtaining continuous latent variables, while others focus on sorting the data into discrete groups or topics [2]. Although both are effective feature reduction methods, the former is mostly used as a way to create new simplified predictive features, while the latter is mostly used as a way to group samples and/or features together [1, 4].

Here we present a discrete latent analysis on the dating profiles of 59946 OkCupid dating profiles in order to try and find good matches for individual users by sorting them into discrete groups. Furthermore, we pay special attention to “Tweeners” (users that don’t identify strongly with any particular group) in order to answer the following question: Is it always a good idea to decrease the number of tweeners by increasing the number of groups “k”?

The report is structured the following way: First, we pre-process the data in order to simplify it and allow us to use it in conjunction with our latent analysis models. This involves the use of “One-Hot” encoding for categorical features, as well as natural language processing for features composed of written responses. Second, we perform Latent Dirichlet allocation on the processed data in order to obtain “k” (k=3, 5, 7, 10, 13, 15, 17, and 20) underlying groups/topics. Third, we study the effects that “k” has on our resulting groups and identify the optimal number of groups needed to characterize the data accurately (reduce number of tweeners). Fourth, we discuss or results and present our conclusions.

2 Related Work and Methods

2.1 Data Description

The data present was obtain from 59946 OkCupid users and 31 features obtained from their dating profiles. The data is a mixture of 10 essays responses, 3 continous variables (age, height, and income), as well as 18 categorical responses (ethnicity, religion, sexual orientation, diet, ext...). Said user profiles were all obtained from the San Francisco Bay Area, and thus represents a niche sample as opposed to a full representation of USA dating preferences. All the data were stripped from confidential information (i.e. names) in order to protect the privacy of the users.

2.2 Data Processing

Data pre-processing was a crucial step in this analysis, as it determined which and what type of variables we are going to be studying. The following is a comprehensive step-by-step procedure (including data pre-processing and interpretation) used to obtain the conclusions presented in this report. The reasoning behind each step is explained in the proceeding paragraphs:

1. Use ”One-Hot” encoding to separate all 18 categorical data samples into individual Boolean responses for each user.
2. Remove the responses that involve continuous responses (age, income, and height)
3. Concatenate each’ users 10 essay responses into a single massive response.
4. Obtain the 100 most used words in all the essays and users (excluding stop words).
5. For each user, create a Bag-Of-Words representation of the top 100 most used words.
6. Combine the previous data with the “One-Hot” encoded data set.
7. Perform Latent Dirichlet Allocation (LDA) and K-Means Clustering on the modified data set (now exclusively composed of multinomial values) in order to identify “k” latent groups.
8. For LDA results, identify users who don’t identify strongly with any particular group
9. Evaluate relationship between “k” (3, 5, 7, 10, 13, 15, 17, 20) and number of “Tweeners”.
10. Compare results for several values of “k” in order to identify the most representative value for this hyper-parameter.

We use “One-Hot” encoding in step 1 because it allows us to convert a complex (yet small) data set into a simpler (yet larger) data set that can be used with LDA without loosing any information. Here, each categorical response for a given question/feature (i.e. religion) is added as a new feature with a value of either “0” or “1” depending on weather a given user is part of said new category (i.e Christian or Buddhist). In Step 2 we remove the continuous variables because these cannot be used with the aforementioned methods.

Steps 3-6 are designed to extract information form the essay responses with out having to do complex natural language processing and/or increase the size of our data set too much. We believe that a BOW representation of the top 100 most used words (excluding stop words) will be sufficient to extract underlying trends in the user essays responses that can then be used to match these users together. For example, we believe that two users who use the word ”art” or ”family” consistently throughout the essays may be good matches in real life.

In step 8, we identified tweeners by finding the users which had the highest median value across their individual group allocation coefficients “z_d” (formally defined in the spotlight section). We used the median as an indicator for the following reasons: 1) the mean for all the users is always 1, and therefore useless (the sum of all LDA coefficients is always 1). 2) Coefficient uniformity (having a low standard deviation) does not always yield a tweener, as users with a single high coefficient still have relatively uniform distributions over its lower coefficient values. 3) The median is not affected by the lowest or highest values, meaning that a high median will always indicate "tweenership" over at least 1 group. We acknowledge that the median is not a perfect metric for this, but we believe that it is sufficient for this study. This **extension** allowed us to include an extra layer of complexity into this study. Finally, Step 9 and 10 are presented at length in the Discussion section.

This resulting processed data sets had the following dimensions: "One-Hot" encoding data set 59946 by 8094, BOW data set: 59946 by 100, Complete data Set (One-Hot + BOW): 59946 my 80194. Given the size of these data sets, we got that each LDA analysis consistently took about 17 minutes to run on a single 2.8GHz Intel Core i5 processor.

2.3 Latent Feature Analysis Models

Our resulting analysis involved the following 2 latent feature analysis methods: Latent Dirichlet Allocation, and K-means Clustering. We chose these because they present two alternate approaches in latent topic analysis: the former is a mixed membership model that allows multiple labels for each users, while the latter is a block-model that identifies hard labels to each. These methods were applied to our data by using the code provided in the SciKitLearn Python Libraries [6] and used with standard values unless specified otherwise.

2.4 Evaluation Criteria

The evaluation of latent features can be tricky and highly subjective: there is no ”labeled” or ”correct” data set when we are talking about these topic identification models. We agree with the argument presented in Blei 2012, which states that there is no reason why a traditional performance metrics obtained from ”held out” data should indicate that we have achieved a correct latent group distribution. A consistent ”log-likelihood” (the logarithm of the probability of obtaining the observed data and the performance metric we use here) across data sets only states that we have a consistent clustering. Furthermore, this only works if we assume that we have a representative data set (our latent groups may not apply to data from the Midwest or the east coast).

3 Spotlight Method: Latent Dirichlet Allocation

Latent Dirichlet Allocation is a generative probabilistic model that assumes that the data that we see (dating profile responses) is generated from a pre-determined number ("k") of hidden variables (i.e. "groups" or "topics"). The method works by assuming that each observable feature is associated with a probability distribution that is unique for each group. While in turn, each group is associated with a distribution within a given user. Therefore, if we knew these distributions a-priori (which we don't) we would be able to approximate the dating profile responses of each user in the data set. The generative join distribution between the groups and responses that results in the probability of a response is written in the following way [1, 3]:

$$p(\beta_{1:k}, \theta_{1:D}, z_{1:D}, w_{1:D}) = \Pi_{i=1}^{k} p(\beta_i) \Pi_{d=1}^{D} p(\theta_i) (\Pi_{i=1}^{k} p(z_{d,n}|\theta_d) p(w_{d,n}|\beta_{1:K}, z_{d,n})$$

Where β_k is the distribution of responses for the "i" group, θ_d is the distribution of groups for a given user "d", z_d are the assigned groups to each user, and w_d are the observed responses for each user "d". Finally, the subscript "n" represents the nth-numbered response. However, as stated before, this is not yet useful to us, as we don't yet know the hidden group distributions. Fortunately, we can calculate the conditional (i.e. posterior) distribution of the hidden groups *given* the observed responses by using their joint probability distribution (the previous equation) and Bayes theorem [3, 1]. This can be written as:

$$p(\beta_{1:k}, \theta_{1:D}, z_{1:D}|w_{1:D}) = \frac{p(\beta_{1:k}, \theta_{1:D}, z_{1:D}, w_{1:D})}{p(w_{1:D})}$$

Where $p(w_{1:D})$ is the marginal probability: the probability of seeing the user's responses given any group distribution. This is equal to sum of all probable joint distributions for a given set of words and "k" topics. Unfortunately, there is no analytical way to calculate marginal probability, meaning that we have to resort to try an approximate it through numerical methods in order to be able to

successfully fit our model. Two examples of these are variational methods, and Gibbs-Markov sampling [1]. An alternative way to present LDA is by the use of a graphical model, which is shown in Fig. 1:

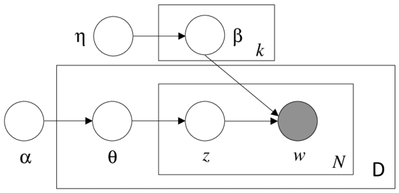


Figure 1: Graphical Model For LDA, where α and η are the hyperparameters used to generate θ and β, respectively. [1]

Figure 1 shows clearly how everything starts and ends with our observed variables (the shaded users' responses). Finally, as it is evident in the two previously-presented equations, there are several assumptions that are embedded within LDA: (1) We assume that user responses are independent of each other, (2) that groups are independent of each other, and (3) that we know the number of latent groups before doing the analysis (i.e. k is a hyperparameter)

4 Results and Discussion

After testing LDA and K-means clustering we concluding that the best way forward would be to work exclusively with LDA. This was mainly due to the fact that the time needed to fit the data through K-means clustering was significantly higher than for LDA (1 hour vs 15 min to sort a 59946 vs 8194 data set into 20 groups). Furthermore, for all model fitting we confirmed that the log-likelihood for LDA was consistent between held-out and tested data (not shown here for brevity) [5].

4.1 Discussion: Tweeners

After fitting our LDA model to our pre-proccesed data as a function of "k" we were able to obtain the histograms shown in Fig. 2. These figure clearly shows that as we increase the value of k, we decrease the magnitude of the median z_d coefficients (a proxy for how undecided the model is on labeling a particular user to a particular group) as well as the number of users that exhibit these high median coefficients. This makes intuitive sense: the more groups we have, the easier is to find a match between a group and an individual. Another interesting conclusion we can derive from this figures is the fact that users with large-valued median coefficients (i.e. the tweeners) are normally distributed and the relative height of this distribution decreases as we increase k.

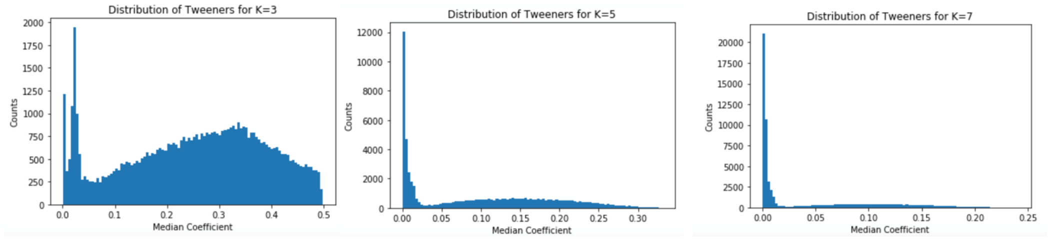


Figure 2: Histogram of Median Coefficient Values of z_d (a proxy for tweeners) as a function of number of groups "k"

We wanted to further investigate the effects of k on tweener prevalance, which is why we decided to plot the average median coefficient for all users as a function of k in Fig. 3. This figure shows that the relationship between these two variables resembles an exponential decay as opposed to a linear dependence. It is clear that, for this data set, the optimal amount of groups is between 7: a value that minimized the magnitude of the median coefficients and number of tweeners, while still having a manageable and representative number of groups.

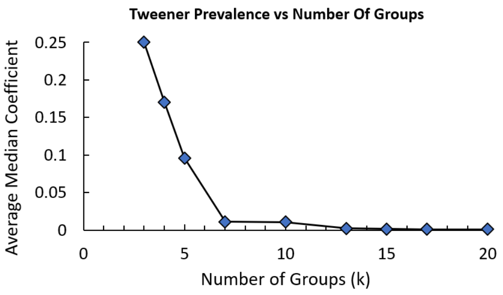
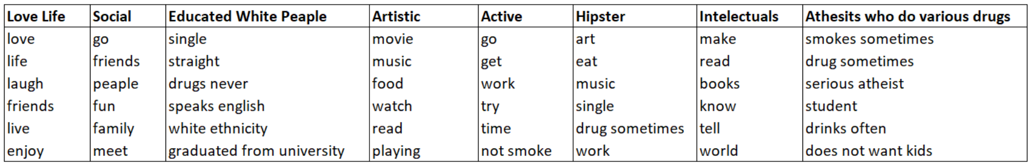


Figure 3: Tweener Prevalance as a function of number of groups “k”.

4.2 Discussion: Grouping Users Together

Having identified the optimum number of groups that minimize the number of users that are not clearly assigned to any group, we then proceeded to label these groups in order to see if the resulting group distributions were coherent. Furthermore, we identified the top features for the remaining tweeners and created an 8th group in order to include them. The results of this analysis is shown in figure 4. If we were in the business of creating positive matches, we would pair the users with the highest z_d coefficients within a particular group (while respecting their gender preferences). This is a straightforward analysis that was skipped here for brevity.



Love Life	Social	Educated White Peaple	Artistic	Active	Hipster	Intelectuals	Athesits who do various drugs
love	go	single	movie	go	art	make	smokes sometimes
life	friends	straight	music	get	eat	read	drug sometimes
laugh	peaple	drugs never	food	work	music	books	serious atheist
friends	fun	speaks english	watch	try	single	know	student
live	family	white ethnicity	read	time	drug sometimes	tell	drinks often
enjoy	meet	graduated from university	playing	not smoke	work	world	does not want kids

Figure 4: Labeled groups and their top features for k=7. The the tweener group is last on the RHS.

The labels and features in Figure 4 show that LDA can be a useful tool when trying to group people together. Note however, that the group labels are a subjective interpretation of the extracted features. Furthermore, we found it really interesting that the tweener group had a strong correlation to users who are atheists-that-do-various-drugs-and-are-students. It seems like these users many share too many (or too few) features with other groups (i.g. hipsters and intellectuals) and thus cannot be sorted successfully.

5 Conclusions

In this report, we used Latent Dirichlet Allocation to successfully sort 60,000 OKCupid dating profiles into 8 individual groups. In order to do this, we identified the optimal number of groups “k” = 7 that produced interpretable and representative latent groups while also minimizing the number of tweeners (users with ambiguous group labels). We found this optimum“k” by investigating the magnitude and prevalence of tweeners as a function of it. The resulting groups were: 1) intellectuals 2) educated white people 3) artistic people, 4) active people, 5) hipsters, 6) people who love life, and 7) social people. We also identified that tweeners tend to be part of an 8th group which consists of ”drug-loving-students-atheists”.

Future directions of this project include: 1) using modified LDA models that can reduce our dependence on the three previously-mentioned assumptions [1]. 2) Comparing our data to actual dating results, which would allow us to obtain traditional evaluation metrics such as accuracy, precision, recal, and a Receiver Operating Characteristic. 3) We could use complex language processing methods (distributional approaches) to extract a better representation of the essay responses.

References

[1] David Blei, Lawrence Carin, and David Dunson. Probabilistic topic models. *IEEE Signal Processing Magazine*, 27(6):55–65, 2010.

[2] Otis Dudley Duncan, Magnus Stenbeck, and Charles J. Brody. Discovering Heterogeneity: Continuous Versus Discrete Latent Variables. *American Journal of Sociology*, 93(6):1305–1321, may 2002.

[3] Barbara Engelhardt. Lecture 18: Probabilistic Topic Models, 2019.

[4] An Gie Yong and Sean Pearce. A Beginner's Guide to Factor Analysis: Focusing on Exploratory Factor Analysis. Technical Report 2, 2013.

[5] Jonathan Lu. Sample Analysis of OKCupid Data¶, 2019.

[6] Pedregosa Fabian, Vincent Michel, Grisel OLIVIER, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Jake Vanderplas, David Cournapeau, Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Bertrand Thirion, Olivier Grisel, Vincent Dubourg, Alexandre Passos, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(Oct):2825–2830, 2011.