Supplementary Information to Predicting and Explaining Behavioral Data with Structured Feature Space Decomposition

Peter Fennell, Zhiya Zuo and Kristina Lerman

S1 Data

We used nine datasets for classification and regression tasks respectively (see Section 4 in the paper) Here we present a more detailed description of the data sources and preprocessing.

S1.1 Data Preprocessing

We downloaded the original data files from UCI Machine Learning Repository [2] and Luís Torgo's personal website[1]. In addition, we conducted further data cleaning before the experiments[2]:

- *Classification*: *(1)* For breast cancer (original), We dropped sample code number columns and removed samples with missing values (all in feature "Bare Nuclei "). *(2)* For parkinsons, we dropped ASCII subject name and recording number.
- *Regression*: *(1)* For appliances energy use, we dropped date columns; *(2)* For residential building, we dropped sales column when the target is costs and vice versa; *(3)* For parkinsons, we dropped columns subject number, age, sex, and test time, which are individual subject information not used for prediction in the original paper [5]. Finally, we note that log transformation of the target values were applied to appliances energy use and both residential building to reduce skewness. Specifically, we used $log_2(x + 1)$ to avoid the logarithm of zero values.

The five remaining human behavior datasets are thoroughly described in the paper.

S1.2 Cross Validation

To evaluate prediction performance, we applied *nested cross validation (CV)* [1] on all datasets (Figure S1). Specifically, we split each dataset into *five* equal-size folds (a.k.a., outer folds; 1 to 5 in Figure S1). Each of the five was picked as test set (i.e., held out from the training and validation process). Given each outer test set (fold 1 in the example), we conducted *inner CV* to find out the best hyperparameters for each predictive model based on the average performance on all four folds (fold 2, 3, 4, and 5.) During *inner CV*, we conducted 4-fold CV, where each of the four folds were held out as validation set (fold 2) and the rest (fold 3, 4, and 5) were inputs to the model. Different sets of *outer CV* may produce different "best" hyperparameters. The final prediction performance is the average value of those on each test set.

[1] `https://www.dcc.fc.up.pt/~ltorgo/Regression/DataSets.html`

[2] `https://github.com/peterfennell/S3D/blob/paper-replication/data/download-datasets.ipynb`

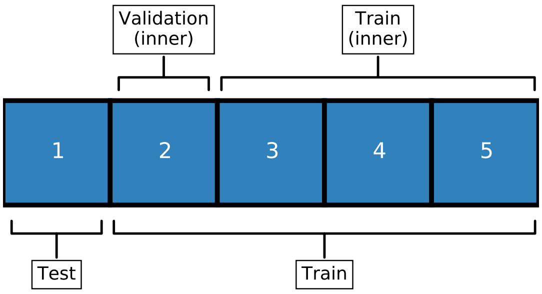


Figure S1: Demonstration of nested cross validation for one iteration.

S1.3 Data Standardization

When applying logistic regressions [3, page 63] and linear support vector machine (SVM) [4], we standardized all features (i.e., subtracting the mean and scaling to unit variance). For all regression data, regardless of regressors, we standardized all columns (i.e., features as well as targets). Note that in both cases, we transformed values in the test set based on the mean and variance of values in the training set.

S2 Visualizing Data with S3D

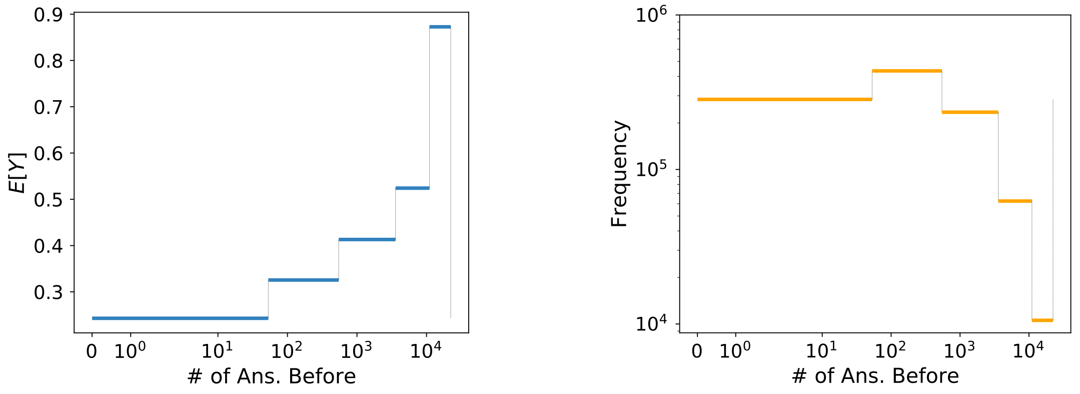
One of the strengths of S3D is its ability to create visualizations of learned models for data exploration. In this section, we provide a detailed description of S3D visualizations of *Stack Exchange* and *Digg* data, starting from the simplest models that partition the data on the most important feature, to the increasingly more complex models that partition the feature space of several most important features.

S2.1 Stack Exchange

The first important feature in Stack Exchange data, *the number of answers before*, reflects users' experience. Intuitively, the more answers a user has written in the past, the more likely that this user is to be active, knowledgeable, and experienced; therefore, the more likely the current answer is to be accepted as best answer by the asker. This is indeed reflected by the model learned by S3D, shown in Figure S2. Evidently, acceptance probability increases monotonically with user experience (Figure S2(a)). On the other hand, most users provide fewer than 1,000 answers. While exceptional activity leads to remarkable acceptance rate, these users are rare in the community (Figure S2(b)).

In the second step, S3D included *signup percentile* as an additional feature to explain acceptance probability (Figure S3). This feature represents user's rank by length of tenure among all other answerers. Overall, we can see an increasing trend of acceptance probability from the bottom left corner to top right, implying a positive relationship where more senior and experienced users are more likely to get their answers accepted as best answers. Indeed, we can see from Figure S3b that senior users are usually those who have written more answers than newcomers. In the mean time, acceptance probability goes down when signup percentile is lower, given the number of answers beyond 3,587. This interesting pattern indicates that answers provided by highly productive users, newcomers, rather than seniors, are more likely to get their answers accepted as best answers.

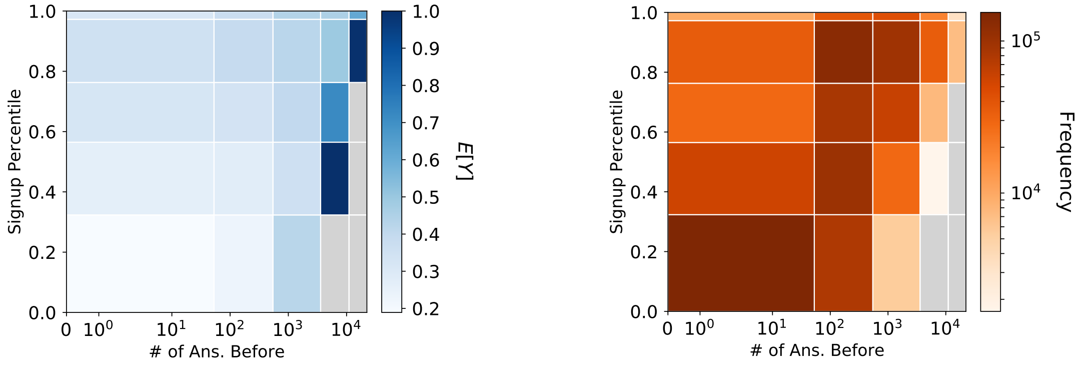
In Figure S4 we proceed to look at models that include the third important feature: *code lines*. Each plot in Figure S4(a) shows a bin with respect to the first important feature (number of answers written



(a) Expected acceptance probability

(b) Frequency

Figure S2: Model of Stack Exchange learned by S3D showing the partition for the first important feature, *the number of answers before*. Each horizontal line in (a) represents average acceptance probability of answers in each bin of the partition. For example, when the number of answers written by the user before the current answer is greater than 10^4 (i.e., the last bin), there is an acceptance probability as high as 0.87; however, relatively few (10,557) samples are in that bin (b), among the lowest of all bins in the partition.



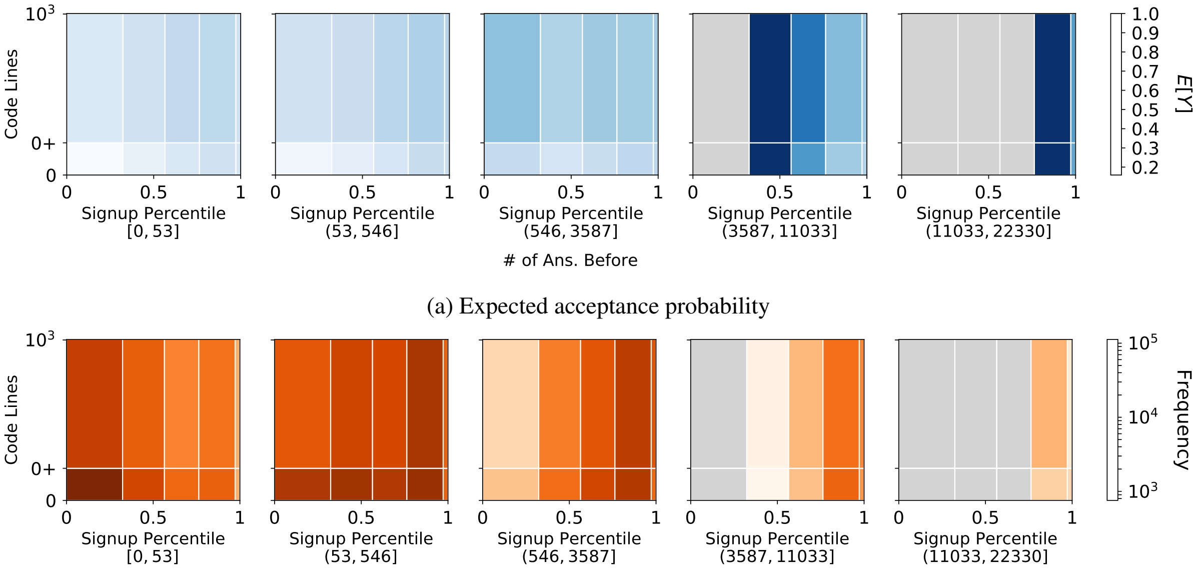
(a) Expected acceptance probability

(b) Frequency

Figure S3: S3D model of Stack Exchange data based on two features, *the number of answers before* and *signup percentile*. Each cell line represents a partition of data based on both features, e.g., when the number of answers written before the current answer ranges from 3,587 to 11,033 and signup percentile is from 0.3 to 0.58, there is an acceptance probability as high as *1*, although the frequency of data samples in this bin is *1,666*.

before this). The range of the bin is shown just under each plot. Note that in each plot there are two zeros on the y-axis. We manually expanded the range $[0, 0]$ from a line to a band to make the binning more clear. In fact, S3D learns from the data that answers that include programming code have a better chance of being accepted (i.e., the color in Figure S4a is *less blue* when the number of code lines is above zero), although most answers do not contain code (Figure S4b). The model with four features can be found in the main paper in Figure 9.

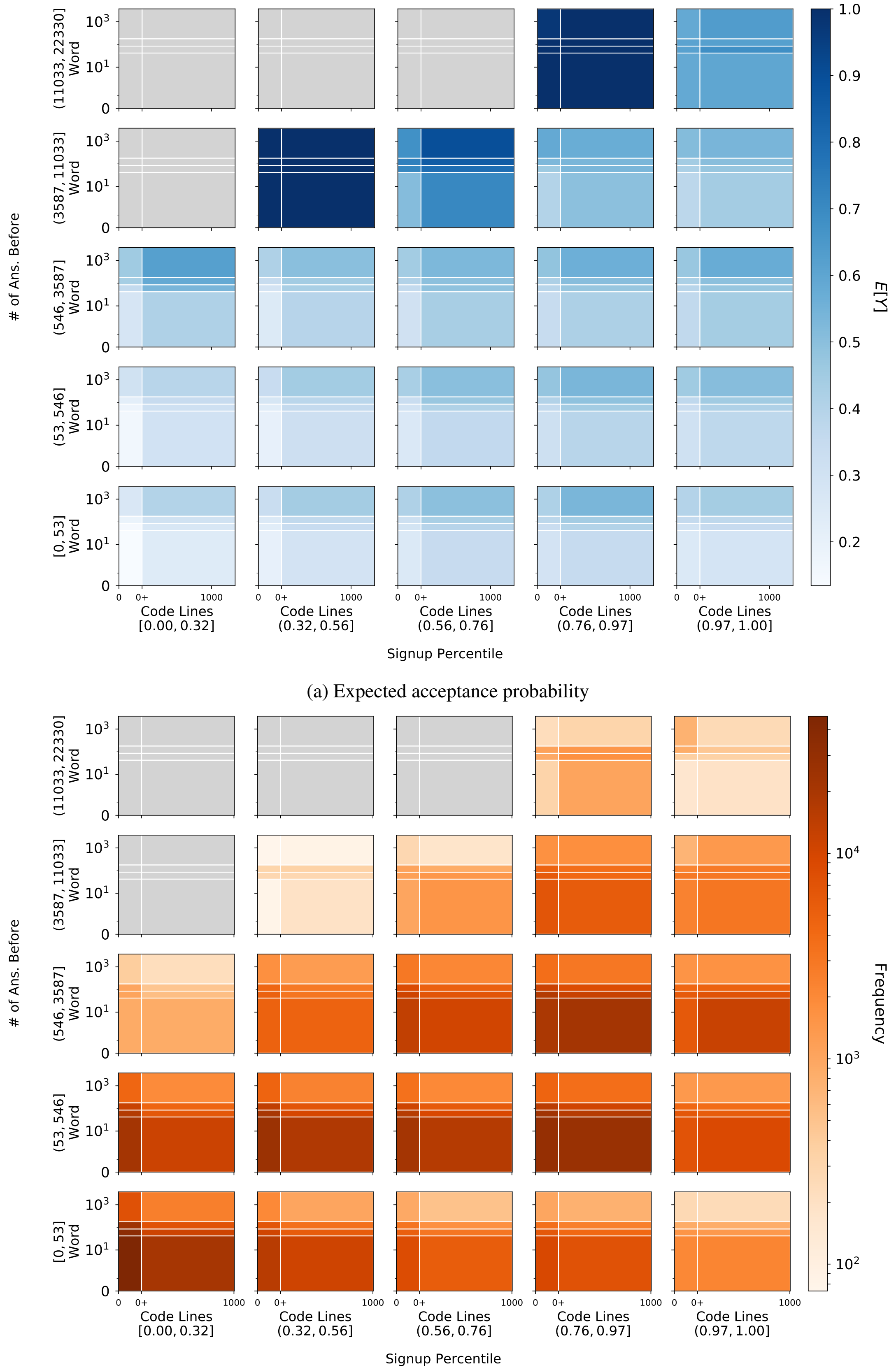
Finally, Figure S5a shows the S3D model with the fourth important feature, *word*, which is the number of words in an answer. Generally, the longer—and presumably the more informative—the answer, the more likely it is to get accepted. Meanwhile, the distribution of words among answers mostly stayed in the bottom half (i.e., below 178 word counts) as shown in Figure S5b. See Section 4.3.2 for an in-depth discussion.



(a) Expected acceptance probability

(b) Frequency

Figure S4: S3D model for Stack Exchange data with three important features: *the number of answers before*, *signup percentile*, and *code lines*. The interpretation is the same as Figure S3.



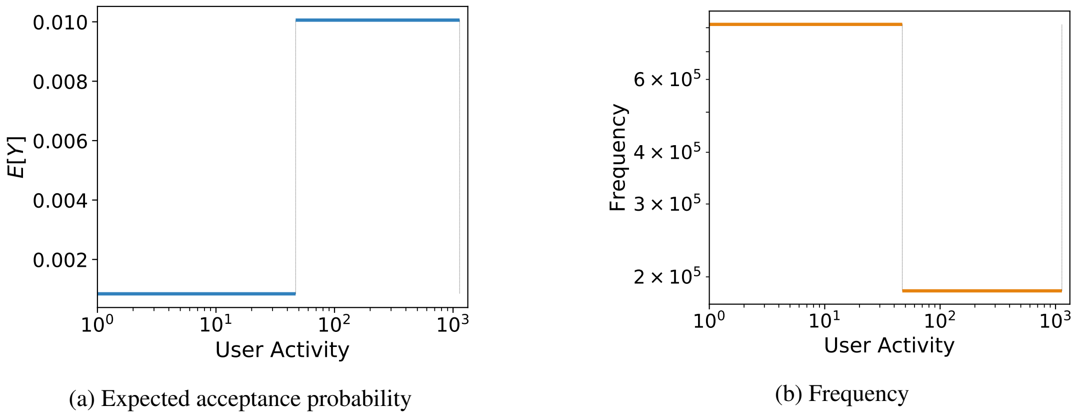
(a) Expected acceptance probability

(b) Frequency

Figure S5: S3D model of Stack Exchange data with four important features: *the number of answers before*, *signup percentile*, *code lines*, and *words*, showing probability of acceptance (top) and number of samples in each bin (bottom).

S2.2 Digg

The first important feature identified by S3D in the Digg data is *user activity* (Figure S6). Recall that the target variable is whether or not a user will "digg" a story a story following an exposure by a friend. Digging a story is similar to retweeting a post on Twitter, hence, we generically refer to it as "adopting" a story or a meme. Generally speaking, more active users are more likely to "digg" stories.

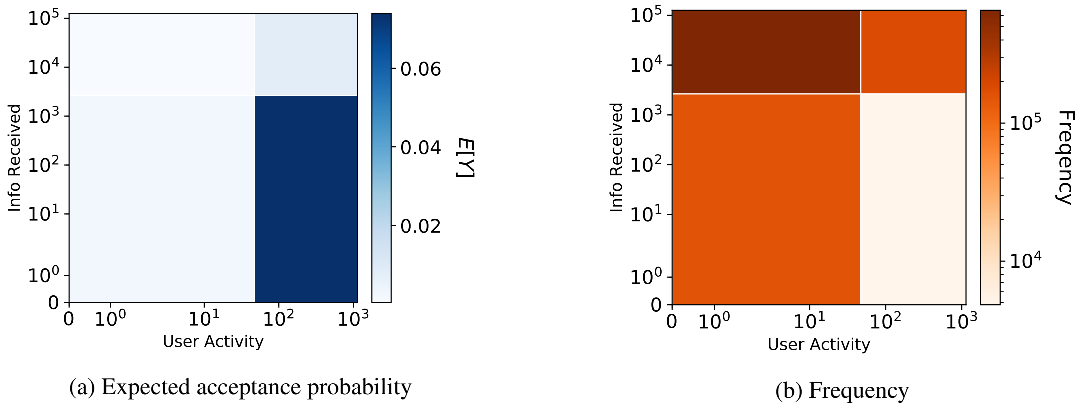


(a) Expected acceptance probability

(b) Frequency

Figure S6: S3D model of Digg data, showing the partition of based on the first important feature *user activity*.

In the next step, S3D finds that *information received by users* is an important feature (Figure S7), with which S3D discovers a more fine-grained partition of data. This feature describes the user's information load, that is the number of stories "dugg" or recommended by friends. Probability to "digg" increases when going from top left to bottom right. Indeed, active users who have lower information load are more likely to "digg" the stories they receive. Nonetheless, most users are not active but receive a huge load of information from their neighbors (Figure S7b). See Figure 10 in the paper for the model with three features.

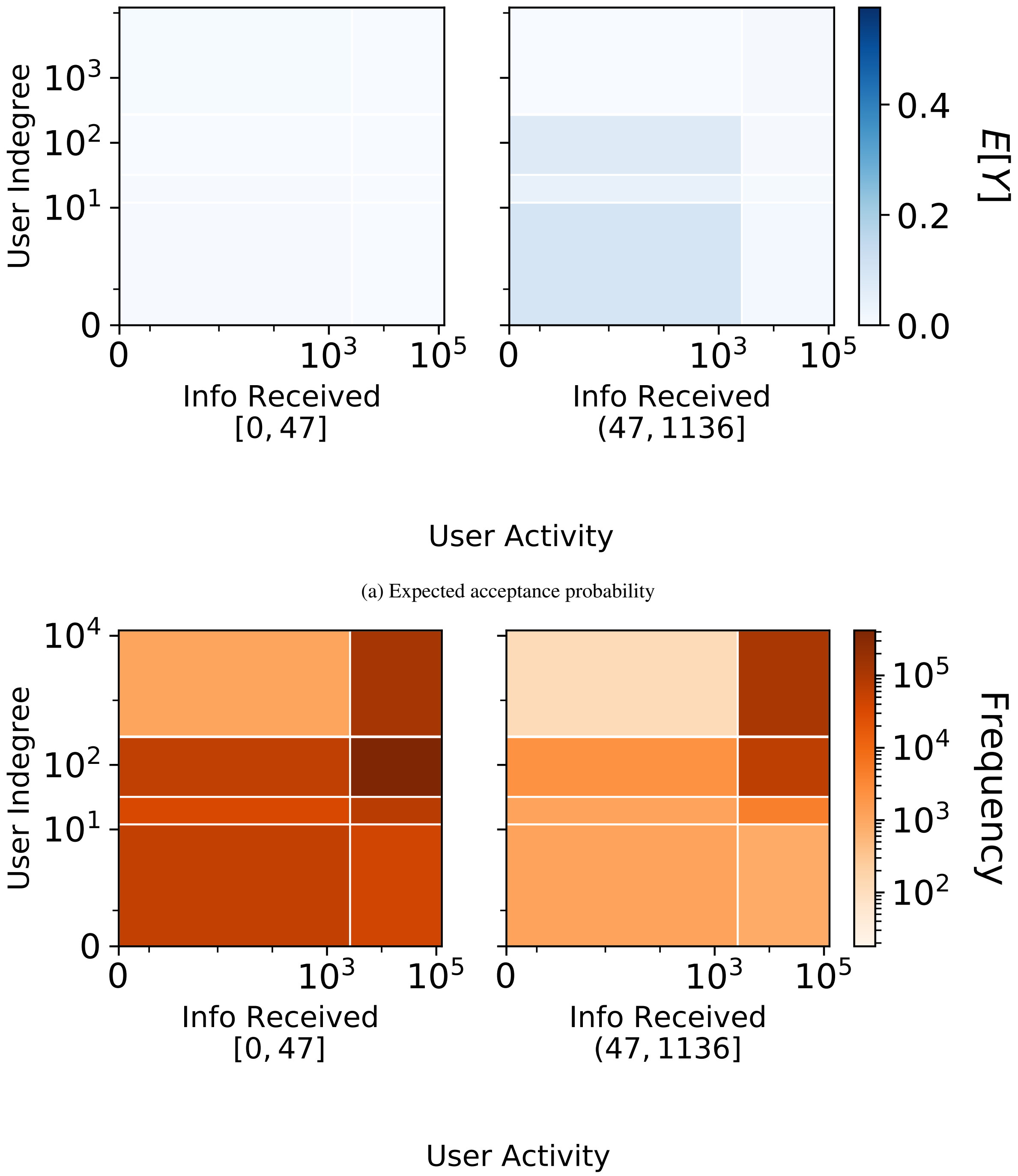


(a) Expected acceptance probability

(b) Frequency

Figure S7: S3D model for Digg with two feature, *user activity* and *information received*.

The third feature that S3D picked is *the current popularity of this meme*, which measures how "viral" a story is (Figure S8a). See Section 4.3.2 for a more detailed description on the learned model. Overall, users tend to "digg " a popular story, implying the *Matthew effect* on diffusion of Digg stories. In Figure S8b, it is interesting to see that most users are exposed to viral stories.



(a) Expected acceptance probability

(b) Frequency

Figure S8: S3D model for Digg data with three important features: *user activity*, *information received*, and *meme current popularity*.

References

[1] Gavin C Cawley and Nicola LC Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11(Jul):2079–2107, 2010.

[2] Dua Dheeru and Efi Karra Taniskidou. {UCI} Machine Learning Repository, 2017.

[3] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York, New York, NY, 2 edition, 2009.

[4] Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, and Others. A practical guide to support vector classification, 2003.

[5] M.A. Little, P.E. McSharry, E.J. Hunter, Jennifer Spielman, and L.O. Ramig. Suitability of Dysphonia Measurements for Telemonitoring of Parkinson's Disease. *IEEE Transactions on Biomedical Engineering*, 56(4):1015–1022, apr 2009.