OptumInsight Data

Ideas

1 Predict the diagnostic codes at a given time

Each patient has a trajectory of medical diagnoses over time. We can consider predicting the probability of observing a medical diagnosis at a future time, based on the past diagnostic trajectory.
We consider the medical diagnoses only at the 3-digit level as we are more interested in the presence of a disease than the specificity of a disease. For example, we are concerned with whether a patient has "Phlebitis and thrombophlebitis" (451). If a patient is diagnosed with 451.11, which represents "Phlebitis and thrombophlebitis of femoral vein (deep) (superficial)", we consider the patient has "Phlebitis and thrombophlebitis" (451), so 451.11 will be truncated to 451.
There are $M = 1101$ unique 3-digit diagnostic codes that appear at least once among our study cohort. Suppose that the medical diagnoses of the i-th patient are recorded at times $T_{ij}, j = 1, \ldots, n_i$. We represent the diagnoses at T_{ij} using a binary vector D_{ij} of length M, where the m-th element d_{ijm} is 1 if the m-th diagnostic code is recorded, and 0 if the diagnostic code is not recorded. For example, patient i diagnosed with 451 and 452 has in D_{ij} two 1's, at the positions denoting 451 and 452 respectively, and 0's at all other 1099 positions. Let the diagnostic history up to time T_{ij} be $\mathcal{H}_{ij} = \{D_{i1}, D_{i2}, \ldots, D_{i,j-1}\}$. We would like to predict either of the following:

(1) $P(d_{ijm} = 1 \mid \mathcal{H}_{ij})$, i.e. the probability of being diagnosed with a particular code m at time T_{ij} given the diagnostic history.

(2) $P(D_{ij} \mid \mathcal{H}_{ij})$, i.e. the joint distribution of diagnostic codes at time T_{ij} given the diagnostic history.

If we are able to obtain 2, then we can obtain the marginal distribution in 1.
Here are some ideas to debate about.

1. Continuous time hidden Markov model. The observation at each time T_{ij} is the multivariate binary diagnostic code D_{ij}. The hidden states may be latent health conditions and the number of hidden states is unknown. We can estimate the number by cross-validation if it is assumed finite, or use the infinite states hidden Markov model (Beal et al., 2001).

2. Discrete time hidden Markov model. Although the medical diagnoses are recorded at irregular time points, we can simplify the model above by discretizing time into one month (or three months). Let T_j be the j-th month after a patient's index VTE date. The observation at T_j is the multivariate binary diagnostic code D_{ij}, where the m-th element d_{ijm} is 1 if the m-th diagnostic code is recorded at least once in month T_j, and 0 otherwise. The hidden states are unknown and can be estimated using the same approach as in 1.

3. Recurrent neural network. The input is the history of multivariate binary diagnostic codes $\mathcal{H}_{ij}$, and the output is the diagnostic code at the next time point $D_{i,j+1}$. RNN takes into account the order of diagnostic codes across time. Different than usual language models using RNN where only one word is predicted at the subsequent position, we need to predict the number of words (diagnostic codes) and the exact words because at each time multiple diagnostic codes may occur.

   Here is a list of related works.

   - (Choi et al., 2016) develops a model Doctor AI that uses longitudinal time stamped EHR data to predict diagnosis, medication categories as well as time for the next visit. It is a temporal model using RNN with diagnostic codes, medication codes or procedure codes as input. The model is implemented with GRU.

     Limitations: Medical events of different types such as ambulatory care visit, outpatient visit, and hospital admission are treated equally. The severity of medical conditions and medications may be different under these circumstances.

- (Bajor and Lasko, 2016) uses RNN with Long Short-Term Memory (LSTM) configuration to predict the likely therapeutic classes of medications that a patient is taking, given a sequence of the last 100 billing codes in their record. But the paper uses one-hot representation on the medical codes and does not take into account the similarity between diseases.
- *cHawkes* model: (Choi et al., 2015) proposes *cHawkes* to construct a global disease network while capturing temporal disease dynamics using Hawkes process. *cHawkes* can also predict when a patient may have specific diseases in the future. cHawkes can control for simple longitudinal patient characteristics by adding a linear term in modeling conditional intensity function of the Hawkes process. Experimentation is done using EHR data, including medical diagnoses, age, and weight.

  Some limitations:

  1. From equation 4, the patient characteristics are modeled via a linear term separate from the disease terms. There may be interaction between patient characteristics and personal disease networks.

  2. The. method only accepts complete cases.

  3. Only simple patient characteristics are included in the experiment, age and weight. We may also want to control for more complex covariates such as sex, races, and prescriptions etc.

  4. Only the primary diagnoses are used in the experiment. We may want to include multiple diagnoses at each measurement. It is also hard for us to define the true primary diagnoses because some patients have several "primary diagnoses".

4. Recurrent neural network with word (diagnostic code) embeddings. In addition to capturing the sequential order of diagnostic codes, we may incorporate similarity between diagnostic codes by embeddings. This requires us to learn features that capture the characteristics of diseases.

2 Predict medications

We can use hidden Markov model to study patient medications (anticoagulants). The hidden states are latent "health conditions" and the number of hidden states can be estimated from the diagnostic code process. The observations are prescribed anticoagulants. Some modeling ideas:

3 Model disease relationship

We can cluster patients based on their diagnostic processes. We need to account for the relationship between diagnostic codes.

1. Model the static network of diagnostic codes.
2. Model the dynamic network of diagnostic codes.
3. Apply language models that account for the hierarchies among words. Some related works:
   - (Morin and Bengio, 2005) builds a hierarchical description of a word by arranging all words in a binary tree where each leaf is associated with one word. Each tree node represents a cluster of words. We can make a probabilistic decision at each tree node to get down to the target word.
   - (Mnih and Hinton, 2008) bases its idea on (Morin and Bengio, 2005), and proposes an automated method for building trees directly from the training data without prior knowledge.

4 Model medical expenditure using diagnostic codes

Medicare Hierarchical Condition Categories (HCC) model was developed to use current year diagnoses and demographics to predict current year healthcare expenditure. The model calculates risk adjustment scores that represent the expected medical costs of a member in the coming year (Formativ Health, 2018), and

that can be used to calculate fees to payers. The risk adjustment factor (RAF) score = demographics (age, gender, living community, insurance plan) + diagnosis (ICD-9, now ICD-10). The medical conditions are hierarchically weighted within the HCC categories so that each patient is coded for only the most severe of the Condition Categories in a group (icd). For example, if a patient has metastatic lung cancer, they will only be assigned the 'CC' for "Metastatic Cancer and Acute Leukemia", and will not be assigned the 'CC' for "Lung and other Severe Cancers". Once the hierarchy rules are applied, the codes are referred to as HCCs. This mapping can change over time. It remained the same from 2007-10. Claims can be declined for a lack of diagnosis specificity.
We can consider modeling the medical expenditure using a marker process. The marks are medical cost, hierarchical diagnostic codes, and places of service.

Relevant literature and methods on modeling the ICD-9 diagnosis codes

Jan 28, 2019

Word Embeddings

A word embedding is a learned representation of document vocabulary. Its purpose is to 1) capture semantic and syntactic relationship between words in a document, and 2) reduce the dimensionality of the vocabulary. Some algebraic operations can be performed on word embeddings to reveal the intrinsic connections between words. For example, the embedding vector representations satisfy "King" - "Man" + "Women" = "Queen".
Here are some popular methods to learn word embeddings.

1. SVD-based methods: We loop over the documents and create some type of count matrix X, and then perform SVD on $X = USV^\top$. We then use rows of U as the word embeddings for all words in the corpus. There are two popular choices of X.

   (1) **Document-term matrix**: We believe that related words tend to appear together in the same documents. Let X_{ij} = the number of times word i appears in document j.
   Issue: the matrix $X \in \mathbb{R}^{|V|\times M}$ is very large, where V is the vocabulary, and M is the number of documents. In addition, X scales with M.

   (2) **Window based co-occurrence matrix**: We hold the same belief as above. Let X_{ij} = the number of times words i and j appear simultaneously within a window of a particular size. Then $X \in \mathbb{R}^{|V|\times|V|}$ and we can analyze X in the following manner.

   a) Perform SVD on $X = USV^\top$.

   b) Select the first k columns of U to get k-dimension word vectors.

   c) The submatrix $U_{1:|V|,1:k}$ is our word embedding matrix. The proportion of variance captured by the first k dimensions is $\frac{\sum_{i=1}^{k}\sigma_i}{\sum_{i=1}^{|V|}\sigma_i}$.

   Issues:

   - As we frequently add new words to the vocabulary, the dimensions of X change very often.
   - X is very sparse because most words do not co-occur.
   - X is high dimensional.
   - It takes quadratic cost to perform SVD.
   - We may need to take into account the imbalance in word frequency in X.

   Because of these issues with SVD-based methods, there are iteration-based methods to learn word embeddings. The most prevalent ones are called word2vec, which is a class of algorithms proposed in (Mikolov et al., 2013).

2. **Skip-gram (SG)**: The goal is to predict the context words given the target (center) word. That is, given the center word w_t at position t, we would like to predict the surrounding words within a window of radius m, which are for example, $w_{t-1}, w_{t-2}, w_{t-3}, w_{t+1}, w_{t+2}, w_{t+3}$. The window size may be one of the parameters of the model. Here we use one-hot representation for the words w_t.

   Let θ be the vector of model parameters. For each word position t, denote $p(w_{t+j}|w_t)$ as the probability of word w_{t+j} given word w_t, where $-m \le j \le m, j \neq 0$ with m being the window size. Our objective function is

$$\max_\theta l(\theta) = \max_\theta -\frac{1}{T}\sum_{t=1}^{T}\sum_{-m\le j\le m, j\neq 0}\log p(w_{t+j}|w_t). \tag{1}$$

Now we specify the unknown quantities in the model to be learned. Let $\mathcal{V} \in \mathbb{R}^{n \times |V|}$ be the input word embedding matrix, where the i-th row $v_i \in \mathbb{R}^n$ is the n-dimensional embedded vector for word w_i. Let $\mathcal{U} \in \mathbb{R}^{|V| \times n}$ be the output word embedding matrix, where the j-th row $u_j \in \mathbb{R}^n$ is the n-dimensional embedded vector for word w_j. Notice that each word w_i has an input representation v_i and an output representation u_i, and we need to learn both representations. We also have $v_i = \mathcal{V}w_i$, and $u_i = \mathcal{U}w_i$.

We measure similarity between words by the inner product, or the so-called cosine similarity. That is, the similarity measure between the embedded vectors u_0 and v_0 is $u_0^\top v_0$. To translate the inner product into probability in (1), we apply the softmax function, i.e.

$$p(w_o|w_t) = \frac{\exp(u_o^\top v_t)}{\sum_{w=1}^{|V|} \exp(u_w^\top v_t)},$$

where w_t represents the center word, and w_o represents the context ("outside") words. Plugging in the embedded vector representations into (1), we can learn the model parameter θ – embedding matrices $\mathcal{V}$ and $\mathcal{U}$ – by for example, gradient descent. Note that $\theta \in \mathbb{R}^{2d|V|}$ if our embedding space is of dimension d.

3. **Continuous bag of words**: We want to predict a center word from the surrounding context (the other way around compare to SG).

   We will use the same notations as in SG, except that we are interested in $p(w_o|w_t)$, where $w_o = (w_{t-m}, w_{t-m+1}, \ldots, w_{t-1}, w_{t+1}, \ldots, w_{t+m})$ with m being the window size. As before, the "o" here means "outside"/context. We need to learn the model parameters – $\mathcal{V}$ and $\mathcal{U}$ – via certain objective function (omitted).

   Model:

   a) For every center word w_t, we average the input embedded vectors to get $\hat{v} = \frac{1}{2m}(w_{t-m} + w_{t-m+1} + \ldots + w_{t-1} + w_{t+1} + \ldots + w_{t+m})$.
   b) Generate a score vector $z = \mathcal{U}\hat{v}$.
   c) Translate the scores into probabilities via the softmax function.
   d) Turn the probabilities into a one-hot vector of the center word.

Mathematical formulation of the problem

Feb 13, 2019

We consider the observations on the study cohort after index VTE date are made at time $t \in [0, \infty)$. We index the patients by $i \in \{1, \ldots, N\}$. Primary observations on the patients consist of diagnosis codes, patient covariates, and an indicator for observable inpatient information. We index the diagnosis codes assigned to patient i at time t by a vector $\boldsymbol{Y}_{it} = (Y_{it1}, \ldots, Y_{itM})^\top \in \{0,1\}^M$ is a multivariate binary vector indicating diagnosis codes and M is the total number of ICD-9 codes. A detailed description about the diagnosis codes can be found on the next page. The bag of ICD-9 codes in the model include all codes at the three, four, and five-digit level. A component Y_{itm} of the vector equals 1 if the m-th ICD-9 code appeared in the patient's record, and 0 if the code did not appear. The model assumes that patients have mixture memberships belonging to a known number of K classes, indexed by $R_{it} \in \{1, \ldots, K\}$. The individual patient membership distribution may vary by time.

We may be unable to observe diagnosis codes on a patient at time t because the patient did not have a medical visit at the time. Hence, we introduce a hidden Bernoulli random variable Z_{it} to indicate the status of observable information. $Z_{it} = 1$ yields observable diagnosis codes and $Z_{it} = 0$ does not emit observable codes. The vector of covariates for diagnosis codes distribution is denoted by $\boldsymbol{X}_{it}$ and has dimension p. The vector of covariates for status indicator is denoted by $\boldsymbol{W}_{it}$, such as length of stay and has dimension q. These two types of vectors may consist of common covariates. We denote the diagnosis history information of patient i up to time t by $\mathcal{H}_{it} = \sigma\{(\boldsymbol{X}_{iu}, \boldsymbol{Y}_{iu}) : 0 \le u \le t\}$, and the inpatient status process up to time t by $\mathcal{F}_{it} = \sigma\{(\boldsymbol{W}_{iu}, Z_{iu}) : 0 \le u \le t\}$.
The data-generating process for patient i at time t is as follows: Unsure about the conditional distributions here.

(1) $Z_{it} \mid \mathcal{F}_{i,t^-}, \boldsymbol{W}_{it} \sim \text{Bernoulli}(p_t)$, where $\mathcal{F}_{i,t^-} = \lim_{s \uparrow t} \mathcal{F}_{i,s}$;

(2) $\theta_{it} \mid \mathcal{H}_{it} \sim \text{Dirichlet}_{K-1}(\alpha_t)$;

(3) $R_{it} \sim \text{Multinomial}(\theta_{it})$;

(4) $\boldsymbol{Y}_{it} \mid \mathcal{H}_{it}, \mathcal{F}_{it}, R_{it} \sim \text{Multinomial}(\theta_t)$.

We are interested in predicting (1) the time to the next observable diagnosis codes and (2) the diagnosis codes observed at the next time.

(a) Suppose the most recent time of observing diagnosis codes is T_0. The next time to the next observable diagnosis codes is

$$T_1 = \inf\{t \ge T_0 : Z_t = 1\}.$$

We are interested in

$$\mathbb{E}[T_1 \mid \mathcal{F}_{T_1}].$$

(b) The diagnosis codes observed at the next time is

$$[\boldsymbol{Y}_{i,T_1} \mid \mathcal{H}_{i,T_1}, \boldsymbol{X}_{i,T_1}, \mathcal{F}_{i,T_1}, R_{i,T_1}],$$

where $[A|B]$ denotes the conditional distribution of A given B.

Proposed priors:

(1) $\boldsymbol{\gamma} \sim \mathcal{N}(0, \sigma_1^2 I)$;

(2) $p_t = \text{logit}\,(\boldsymbol{Z}_i \boldsymbol{\gamma})$;

(3) $Z_{it} \mid \mathcal{F}_{i,t-1}, \boldsymbol{W}_{it} \sim \text{Bernoulli}(p_t)$;

(4) $\alpha_t \mid \alpha_{t-1} \sim \mathcal{N}(\alpha_{t-1}, \sigma_2^2 I)$;

(5) $\theta_{it} \mid \mathcal{H}_{it}, \alpha_t \sim \text{LogisticNormal}(\alpha_t, \sigma_3^2 I)$;

(6) $R_{it} \sim \text{Multinomial}(\theta_{it})$;

(7) $\boldsymbol{Y}_{it} \mid \mathcal{H}_{it}, \mathcal{F}_{it}, R_{it} = k \sim \text{Multinomial}(\theta_{t,k})$.

Parameterization of the Tree-Structured Responses

An ICD-9 code has a rooted tree structure with at most three levels. The root is a three-digit ICD-9 diagnosis code corresponding to a diagnostic category. Its children at the second level are four-digit diagnosis codes indicating diagnostic subcategories. Five-digit diagnosis codes at the third levels show further subclassification of the subcategories. Insurance policies generally require the highest specificity of assigning a code in order for the diagnosis to be reimbursable. Some physicians assign a three-digit code when the diagnosis is unclear at the initial encounter, but they tend to code with the the highest possible specificity for both accuracy and reimbursement. As a result, there are four properties in the observed diagnosis codes in our study cohort. First, if we observe a diagnosis code of a subcategory, we will not observe the code of its parent category. Second, when we observe a diagnosis code whose subcategories should have been observed instead, the exact diagnosis code subcategories are unknown. Third, we expect to see diagnosis codes that are leaves of the code trees more frequently than other non-leaf nodes. Fourth, some diagnosis codes are more similar than others and are likely to co-occur on a patient.

Let a tree of ICD-9 code be $G_m, m \in \{1, \ldots, M\}$, where M is the total number of ICD-9 trees. We index the nodes in tree G_m using $\{Y_j^{(l)}, j \in J_l, l = 1, 2, 3\}$, where l is the index for the level of the node in this tree, and j is the index for the node in level l. Each node $Y_j^{(l)}$ is a Bernoulli random variable, and $Y_j^{(l)} = 1$ if the diagnosis code is assigned to a patient and $Y_j^{(l)} = 0$ if the diagnosis is not assigned. We denote the observed diagnosis codes at time t on patient i using vector $\boldsymbol{y}_{it} = \left(y_{itj}^{(l)}, j \in J_l, l = 1, 2, 3\right) \in \{0, 1\}^{|G_m|}$, where $|G_m|$ denotes the total number of nodes in the tree, $y_{itj}^{(l)} = 1$ if the code is observed and $y_{itj}^{(l)} = 0$ is not observed. Here an unobserved code means that either the patient did not have the disease, or the doctor was unable to diagnose the disease. Given a node in level 1 or level 2 $Y_j^l, l = 1, 2$, let the set of all its children be denoted as $\boldsymbol{Y}_{C(j)}^{l+1}$. In usual directional tree-structured data models, we assume that conditional on a node, all its children are independent. However, we do not make this assumption because as the disease categories are correlated. Instead, we assume that the conditional distribution of children $\boldsymbol{Y}_{C(j)}^{l+1}$ given the parent Y_j^l is a multivariate Bernoulli distribution with probability vector $\boldsymbol{\pi}_{C(j)}$ following a Dirichlet distribution. In addition, we assume that given all nodes at a lower level, the respective collections of children are conditionally independent. Taking into account the four properties of the tree-structured diagnosis codes mentioned above, we describe the data-generating process as the following.

- $\boldsymbol{\pi}_{C(j)} \sim \text{Dirichlet}_{|C(j)|}\left(\boldsymbol{\alpha}_{C(j)}\right)$;
- $\left[\boldsymbol{Y}_{C(j)}^{l+1} \mid Y_j^l\right] \sim \text{Bernoulli}_{|C(j)|}\left(\boldsymbol{\pi}_{C(j)}\right)$;
- $P\left(\boldsymbol{Y}_{C(j)}^{l+1} = \boldsymbol{y}_{C(j)}^{l+1} \mid Y_j^l = 0\right) = 0$, for any vector of observed values $\boldsymbol{y}_{C(j)}^{l+1}$;
- $\boldsymbol{Y}_{C(j)}^{l+1} \perp\!\!\!\perp \boldsymbol{Y}_{C(k)}^{l+1} \mid Y_j^l, Y_k^l$ for any nodes j, k in level 1 or level 2.

Using this parameterization of the responses, we need to estimate parameter vectors $\boldsymbol{\alpha} = \left\{\boldsymbol{\alpha}_{C(j)}, j \in J_l, l = 1, 2\right\}$.

Relevant Literature

Margaret E. Roberts (2016) proposed a hierarchical mixed membership model called Structural Topic Model (STM) that models both topic prevalence and topic content with different covariates. The model allows for time-varying topic proportions if time is included as a covariate, but does not allow for repeated measurements on the covariates.
Blei and Lafferty (2006) proposed the Dynamic Topic Model (DTM) that allows topic proportions of documents to vary over time. This is done by dividing data by time slice, and introducing a Markov chain on the mean of the topic parameters and word parameters, respectively. Different than latent Dirichlet allocation, the Dirichlet distribution on topics is replaced with normal distribution. Topic proportions do not change over time while individual document membership distribution does.

References

`icd` package R documentation. `https://www.rdocumentation.org/packages/icd/versions/2.4.1/topics/icd9_map_hcc`. Accessed: 2018-12-30.

Bajor, J. M. and Lasko, T. A. (2016). Predicting medications from diagnostic codes with recurrent neural networks.

Beal, M. J., Ghahramani, Z., and Rasmussen, C. E. (2001). The infinite hidden markov model. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, NIPS'01, pages 577–584, Cambridge, MA, USA. MIT Press.

Blei, D. M. and Lafferty, J. D. (2006). Dynamic topic models. In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, pages 113–120.

Choi, E., Bahadori, M. T., Schuetz, A., Stewart, W. F., and Sun, J. (2016). Doctor ai: Predicting clinical events via recurrent neural networks. In Doshi-Velez, F., Fackler, J., Kale, D., Wallace, B., and Wiens, J., editors, *Proceedings of the 1st Machine Learning for Healthcare Conference*, volume 56 of *Proceedings of Machine Learning Research*, pages 301–318, Children's Hospital LA, Los Angeles, CA, USA. PMLR.

Choi, E., Du, N., Chen, R., Song, L., and Sun, J. (2015). Constructing disease network and temporal progression model via context-sensitive hawkes process. In *Proceedings of the 2015 IEEE International Conference on Data Mining (ICDM)*, ICDM '15, pages 721–726, Washington, DC, USA. IEEE Computer Society.

Formativ Health (2018). Understanding hierarchical condition categories (hcc). Technical report.

Margaret E. Roberts, B. M. S. . E. M. A. (2016). A model of text for experimentation in the social sciences. *Journal of the American Statistical Association*, 111(515):988–1003.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Mnih, A. and Hinton, G. (2008). A scalable hierarchical distributed language model. In *Proceedings of the 21st International Conference on Neural Information Processing Systems*, NIPS'08, pages 1081–1088, USA. Curran Associates Inc.

Morin, F. and Bengio, Y. (2005). Hierarchical probabilistic neural network language model. In *AISTATS*.