ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

Propuesta de proyecto:
Sleepy Bayesian

Fernanda Weiss - 18406890
Ignacio Espinoza - 18406882

IIC3695 Tópicos Avanzados en Inteligencia de Máquina

Profesor: Karim Pichara Baksai

Ayudantes : Francisco Pérez Galarce, Orlando Vásquez

1. Definición problema

Descansar lo suficiente es importante para llevar una vida saludable, pues durante el sueño el cuerpo realiza procesos importantes que nos permiten mantenernos sanos, por ejemplo se relaja la columna, se restauran los músculos, se regulan los niveles de azúcar y apetito, entre otros efectos[1]. Por lo tanto, malos hábitos de sueño, como dormir poco, dormir mucho, o dormir sin un patrón estable de sueño [2] aumenta el riesgo de sufrir enfermedades de distinta gravedad, desde enfermedades leves como mayor probabilidad de desarrollar un resfriado hasta enfermedades crónicas como problemas al corazón, diabetes, obesidad, etc. Además, esta actividad nos ayuda en otras tareas. Por ejemplo, el cerebro fija los recuerdos y lo aprendido durante el día, repone del cansancio para poder realizar las funciones óptimamente como concentración, pensar con claridad y reaccionar rápido, pues hemos visto accidentes de tránsito que ocurren por la falta de sueño.

Debido a lo explicado anteriormente es que consideramos importante que cada persona pueda obtener su patrón de sueño, pues si presenta alguna enfermedad con esta información un especialista tendría más antecedentes para detectar dicha enfermedad y también para saber cómo enfrentarla. No obstante, de igual modo esta información puede ser útil para que uno haga uso de ella, pues podrá regular cuánto duerme cada día y cuándo le es más provechoso dormir. Esta información para estudiantes como nosotros sería beneficiosa, ya que al no tener una rutina clara con los diferentes horarios y quehaceres no hay un claro patrón de sueño, lo que claramente nos lleva a enfermarnos, y utilizando los resultados de este estudio uno se podría organizar mejor para cuidar nuestra salud.

Ya existen aplicaciones como Sleep Calculator[2] para cuidar nuestra higiene de sueño [1], la cual solo calcula nuestro ciclo de sueño y nos ayuda a determinar cuándo es el mejor momento para ir a dormir o cuando despertar para sentirnos descansados. Sin embargo, ésta aplicación no es personalizada, ya que funciona para el ciclo de sueño de una persona “normal”.

Estudiar este patrón de sueño que indique cuando deberíamos dormir y cuando deberíamos despertar según nuestra rutina podría ser relevante para posibles implementaciones en nuestra vida diaria. Por ejemplo, saber cuán probable es que una persona se quedará dormida al volante dependiendo de la hora en que quiere manejar, a qué hora estaremos realmente despiertos para rendir un examen, tener

[1] http://www.bbc.com/mundo/noticias-41811949
[2] http://sleepcalculator.com/

un despertador inteligente que sepa cuándo sería bueno despertarnos si es que aun dormimos, entre otros.

El tracker de actividad inteligente Vivosmart HR posee la capacidad de medir la frecuencia cardiaca en la muñeca las 24 horas del día, durante los 7 días de la semana. Entrega información de calorías quemadas y cuantifica la intensidad de las actividades físicas. Estudiando el ritmo cardíaco, junto con el trackeo del movimiento, se puede determinar cuando una persona se queda dormida y se despierta.

El proyecto está enfocado al estudio de datos del sueño para crear un modelo que determine la probabilidad a posterior de que una persona se duerma en un determinado horario. A su vez, determinar la probabilidad a posterior de que una persona se despierte. Ambas probabilidades se determinan en base al historial de sueño de una persona en específico.

Como el tiempo es una variable continua, calcular la distribución a posterior a cabalidad no es posible analíticamente. Por esto es necesario utilizar métodos para simular la distribución que modela el tiempo. Dentro de esos métodos se encuentran los algoritmos de *Markov Chain Monte Carlo (MCMC)*. MCMC son métodos numéricos usados para calcular, aproximar y simular expresiones o sistemas matemáticos complejos y difíciles (a veces imposibles) de evaluar.

2. Características de los datos

Los datos a utilizar fueron recopilados por William Koehrsen[3] por medio de su trackeador VivoSmart. Las mediciones cardíacas fueron transformadas a información que indican el momento que se durmió y despertó respecto a la fecha de medición.

Se trabajará con el dataset sleep_wake.csv que cada una de sus entrada marca el minuto, desde las horas habituales de dormir y despertar mencionadas anteriormente, en la que el tracker detecta que la persona se durmió y el momento en que esta despertó. Además, se registra el día de la medición. En total el dataset tiene 64 registros, donde cada uno representa un día de mediciones. Por lo tanto, esta información se debe procesar para obtener datos manejables entre las transiciones despierto-dormido y dormido-despierto. Así, se crearán dos datasets:

[3] https://medium.com/@williamkoehrsen

uno que describa los datos cuando la persona se va a dormir y el segundo cuando la persona se vaya a despertar.

Puesto que se manejan pocos datos será difícil que los modelos a entrenar puedan predecir bien datos futuro pues no podrán generalizar bien y existirá overfitting. La solución a esto se hará un preprocesamiento de datos que permita agregar información real a cada dataset. Se tomará como punto de inflexión la hora común de ir a dormir (10 p.m) para la transición despierto dormida y como hora común de despertar (6 a.m) para la transición inversa. Con este punto de inflexión se generarán tantas filas por minutos, en un intervalo de tiempo alrededor a este, para analizar cuánto es que varía su comportamiento de sueño, así se podrán obtener distintos dataset. Por ejemplo, solo analizar +/- 1 hora desde el punto de inflexión, y por otro lado analizar +/- 2 horas desde el punto de inflexión. Cada fila luego del preprocesamiento de los datos de entrada indican la hora y fecha en que fueron tomadas, una variable booleana que indica si la persona estaba dormida o despierta y un valor de offset indicando la cantidad de minutos de diferencia con la hora habitual en que él se duerme (10 p.m.) y con la hora que habitualmente se despierta (6 a.m.). Así, al agregar +/- 1 hora se tendrán 7.680 para cada dataset (15.360 en total), y con +/- 2 horas 15.360 datos para cada dataset (30.720 en total).

3. Modelos a utilizar

Dado que target que buscamos es binario, pues queremos saber si está durmiendo o despierto, y buscamos una transición suavizada entre los dos estado (durmiendo y despierto), se debe encontrar una función que modele la probabilidad de transición entre estados. Para resolver esta tarea postulamos una regresión logística[3], cuyo dominio es [0,1] con una transición lineal entre ambos límites.

$$\pi(\beta) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}} \qquad (1)$$

donde β contiene a β_0 y β_1. β_0 controla el shift de posición de la función y β_1 indica la dirección. En ambos extremos tiene colas pesadas que aseguran que datos muy alejados tomen probabilidades aproximadas a cero (por la izquierda) como aproximadas a uno (por la derecha), como lo muestra la Imagen 1.

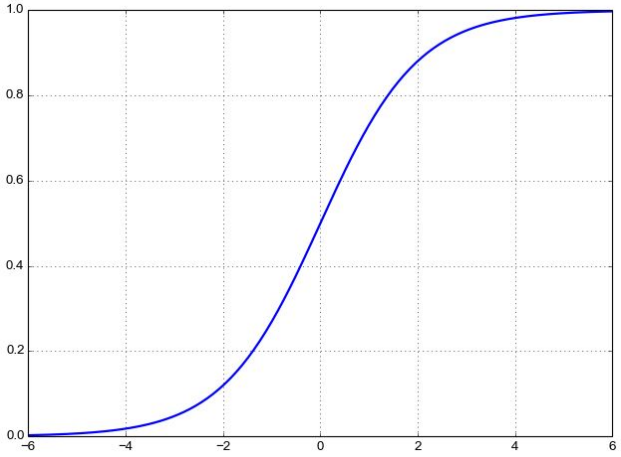

Imagen 1: Función Logística

Como se mencionó el target *y* es una variable binaria, la cual se modela mediante una distribución Bernoulli con probabilidad dada por la función logística con parámetro β:

$$\hat{y} \sim Ber(\pi(\beta))$$

Por lo tanto, nuestra likelihood está dada por:

$$P(x,y \mid \beta) = \prod_{n=1}^{N} \pi(\beta)^{y_n}(1-\pi(\beta))^{(1-y_n)} = \prod_{n=1}^{N} \left(\frac{1}{1+e^{-(\beta_0+\beta_1 x)}}\right)^{y_n}\left(1-\frac{1}{1+e^{-(\beta_0+\beta_1 x)}}\right)^{(1-y_n)}$$

donde N es la cantidad de datos.

Luego, definimos el prior del parámetro β. Asumiremos que será una Gaussiana bivariada con parámetro μ y Σ, que para nuestro modelado se comportan como hiper-parámetros.

$$P(\beta) = N(\beta \mid \mu, \Sigma) = \frac{1}{\sqrt{2\pi\,|\Sigma|}} e^{-\frac{1}{2}(\beta-\mu)^T \Sigma^{-1} (\beta-\mu)}$$

Así, la posterior que nos interesa está dada por:

$$P(\beta \mid D) = P(\beta \mid x,y) = \frac{P(x,y \mid \beta)P(\beta)}{P(x,y)} = \frac{P(x,y \mid \beta)P(\beta)}{\int P(x,y)}$$

Tomando la posterior como:

$$P(\beta \mid x,y) \propto P(x,y \mid \beta)P(\beta) = \prod_{n=1}^{N} \left(\frac{1}{1+e^{-(\beta_0+\beta_1 x)}}\right)^{y_n} \left(1-\frac{1}{1+e^{-(\beta_0+\beta_1 x)}}\right)^{(1-y_n)} \frac{1}{\sqrt{2\pi\,|\Sigma|}} e^{-\frac{1}{2}(\beta-\mu)^T \Sigma^{-1} (\beta-\mu)}$$

Como se puede apreciar, nuestra posterior no tiene una forma cerrada y obtener su denominador es demasiado caro. Por lo tanto, para poder estimarla utilizaremos diferentes algoritmos de MCMC para simular parámetros β que mejor describan los datos. Estos algoritmos nos permitirán sacar muestras tanto de la distribución a posteriori de que una persona se duerma o se despierte. Así, con dichas muestras se podrá estimar el valor óptimo de los parámetros.

4. Evaluación de los resultados

Se comparará los resultados de los tres algoritmos a utilizar: Metropolis Hastings, Gibbs Sampling, Simulated Annealing y Slice Sampling, los cuales será evaluados bajo el criterio de convergencia a su distribución estacionaria, que será calculado mediante el método Gelman-Rubin, Running means, y mediante métodos gráficos.

Primero, se realizará un pre-procesamiento de los datos junto con su visualización para comprender la naturaleza de estos. Luego, se realizará un conjunto de pruebas para cada algoritmo usando diferentes configuraciones para generar las muestras. Se visualizarán y analizarán las muestras obtenidas, para determinar los distintos efectos que pueden tener los parámetros sobre los modelos de los datos. Finalmente, se probará la convergencia de cada algoritmo con los métodos anteriormente nombrados.

Referencias

[1] Manrique, J. J. (2011). Higiene del sueño. Higiene, 39(3).
[2] Bernardo, Á. (2013, October 03). Efectos del sueño: Dormir mal puede provocar problemas en la salud. Recuperado de https://hipertextual.com/2013/10/efectos-sueno-corazon-diabetes
[3] Acquah, H. D. (2013). Bayesian Logistic Regression Modelling via Markov Chain Monte Carlo Algorithm. Journal of Social and Development Sciences, 4(4), 193-197.