



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

Tarea 2: Tópicos Avanzados en Inteligencia de Máquina

Profesor : Karim Pichara Baksai

Ayudantes : Ignacio Becker, Francisco Pérez Galarce, Matías Vergara

Fecha de entrega : 24 de Mayo 2019, 23:59

1 Descripción de la actividad

Esta tarea consiste en estudiar y analizar una aplicación de dos de los métodos de muestreo vistos en el curso: (*Gibbs Sampling* y *Metropolis-Hasting*). Posteriormente, dichas muestras serán utilizadas para la generación de series de tiempo artificiales de estrellas que exhiben un comportamiento particular, denominadas *estrellas variables*. Estas simulaciones podrían utilizarse para obtener más datos de entrenamiento, por ejemplo para entrenar clasificadores basados en redes neuronales.

2 Contexto de la aplicación

El modelo que utilizaremos para representar una serie de tiempo $x(t)$ es el siguiente:

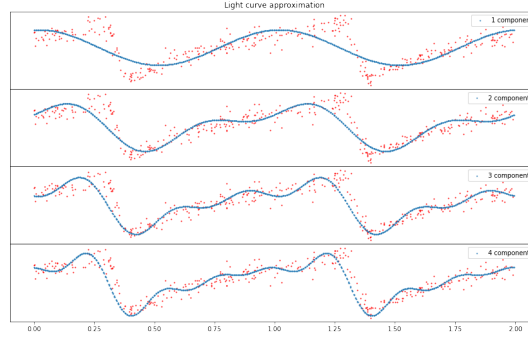
$$x(t) = \sum_{i=0}^N a_i \sin(2\pi(i+1)ft) + b_i \cos(2\pi(i+1)ft)$$

Como pueden notar, para conocer (y graficar) una serie de tiempo necesitamos sus parámetros: $\{a_0, a_1, b_0, b_1, f\}$.

Las series de tiempo astronómicas son complejas dado que el muestreo es aleatorio, son ruidosas y generalmente tienen distinta cantidad de observaciones. Para poder aplicar herramientas de *machine learning*, se extraen descriptores a partir de los datos. Los parámetros del ajuste son un pequeño subconjunto de los descriptores que se usan en la actualidad.

En el gráfico se muestra el ajuste usando el modelo anterior considerando de $N = 1$ a $N = 4$ componentes, es decir, el límite superior de la sumatoria se hizo variar desde 1 hasta 4. Con una componente, el modelo sólo logra replicar el comportamiento oscilatorio. A medida que se agregan más elementos a la sumatoria, se ajusta con más precisión a los datos, pudiendo reconstruir con gran exactitud el comportamiento de una serie de tiempo que representa la variabilidad del brillo de una estrella variable periódica.

Figure 1: Ejemplos series de tiempo graficadas con el modelo paramétrico $(\sin(x), \cos(x), x)$



3 Instrucciones

Volviendo al modelo generador de series de tiempo, consideremos un modelo $m^*(\alpha)$ que depende de los parámetros $\alpha = \{a_0, a_1, b_0, b_1, f\}$. Este modelo corresponde a una dos mezclas de Gaussianas en tres dimensiones. Las dimensiones de la gaussiana 1 y 2 corresponden a $\{a_0, b_0, f\}$ y $\{a_1, b_1, f\}$ respectivamente. Debido a dependencias físicas del modelamiento (f aparece en las dos gaussianas). En otras palabras, necesitamos que en cada muestra i , $\alpha^i = \{a_0^i, a_1^i, b_0^i, b_1^i, f^i\}$, la dimensión f de cada mezcla gaussianas sea consistente.

Para simplificar el problema en el archivo *Params_ecl.npz* les entregamos las covarianzas y pesos de cada mezcla de Gaussianas obtenidos del ajuste de datos reales. Para todos los experimentos, tenga en cuenta que la frecuencia no puede ser menor a 0. El archivo está serializado en *numpy*.

Considerando los datos entregados, encuentre las medias de las dos mezclas de gaussianas utilizando el método *Metropolis-Hasting*. Para esto, recomendamos seguir los siguientes pasos:

- Implemente la función likelihood de las mezclas de gaussianas.
- Implemente *Metropolis-Hasting* justificando la definición de todos los elementos asociados (Q^* , Burn-in, Sample window, etc.).
- Evalúe la convergencia de la cadena, para ello necesitará realizar una pequeña revisión de los métodos de convergencia disponibles, puede utilizar alguna implementación disponible o implementar el método.

Ahora que tiene todos los parámetros de las mezclas de gaussianas, se solicita utilizar el método *Gibbs Sampling* para la generación de curvas de luz de acuerdo al modelo presentado en la sección 2. Visualice un conjunto de series de tiempo obtenidas desde el muestreo.

¿Es posible generar series de acuerdo a su frecuencia de aparición en los datos originales? ¿Cuándo convendría generar series poco frecuentes en la práctica? Comente al respecto.

Usted es libre de realizar todos los supuestos que estime necesarios para el desarrollo de la tarea.

4 Entrega

- La entrega debe ser realizada en un .zip con todos los archivos necesarios. No incluya los datos.
- El archivo de entrega debe ser subido al cuestionario abierto en el sistema SIDING específicamente para esta tarea hasta el día y hora señalada con el nombre `[numero_alumno].T2`.
- Las tareas atrasadas enviadas por otro medio no serán evaluadas y serán calificadas con nota 1.0.
- La tarea es estrictamente individual.
- El documento principal debe ser un jupyter notebook con el código. Además deben subir un archivo con extensión `.py` cuyo contenido es **solamente el código** (el mismo del `.ipynb`). El archivo `.py` deberán nombrarlo `[numero_alumno].T1.py`.
- Cualquier instrucción adicional y necesaria para la revisión debe ser escrita en un archivo README.txt contenido en el .zip

5 Política de Integridad Académica

Los alumnos de la Escuela de Ingeniería deben mantener un comportamiento acorde al Código de Honor de la Universidad:

“Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad”

En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un procedimiento sumario. Ejemplos de actos deshonestos son la copia, el uso de material o equipos no permitidos en las evaluaciones, el plagio, o la falsificación de identidad, entre otros. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica en relación a copia y plagio: Todo trabajo presentado por un alumno (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), sin apoyo en material de terceros. Si un alumno (grupo) copia un trabajo, se le calificará con nota 1.0 en dicha evaluación y dependiendo de la gravedad de sus acciones podrá tener un 1.0 en todo ese ítem de evaluaciones o un 1.1 en el curso. Además, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir un procedimiento sumario. Por “copia” o “plagio” se entiende incluir en el trabajo presentado como propio, partes desarrolladas por otra persona. Está permitido usar material disponible

públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente.