

Práctica 7: Introducción a Cadenas de Markov Ocultas

Semana 7

Esta práctica está basada en el material disponible en el sitio web de la Ecole Polytechnique Fédérale de Lausanne, para el curso de Procesamiento y Reconocimiento de Habla. Se utilizarán como datos el archivo `data.mat`, y los programas de Matlab que figuran en el campus. Antes de utilizar los programas debe correrse el script que figura en el campus como `inic_hmmm.m`, que no hace otra cosa que cerrar todas las figuras, limpiar las variables, inicializar los colores de gráficos, y cargar los datos.

1. Descripción de un modelo de Markov con gaussianas.

- Las variables `hmm1` a la `hmm6` son variables tipo estructuras, que contienen en su primer campo las medias de las gaussianas de cada estado, en el segundo las varianzas, y en el tercero la matriz de transición. De acuerdo al valor que poseen dichas variables ya prefijados, determine para cada uno de ellos el diagrama de estados correspondiente incluyendo la red de transiciones posibles.
- Utilizando Matlab, genere las curvas de nivel de las funciones de distribución de cada estado, de acuerdo a las medias y varianzas dadas.
- La función `genhmm.m` genera las características 2-D (valores de las variables aleatorias emitidas durante la sucesión de estados de la cadena) correspondientes a un dado modelo de Markov de la forma de los que contiene `data.mat`. Para cada uno de los modelos `hmm1` a `hmm6` suministrados en `data.mat`, genere una emisión de características, y gráfiquelas utilizando `plotseq.m` y `plotseq2.m`. Verifique por inspección que las emisiones son posibles, contrastando los valores que van tomando las características con lo esperable a partir de los parámetros de los modelos (medias, varianzas y transiciones).

2. Entrenamiento de parámetros para Cadenas Ocultas (Algoritmo de Baum-Welch)

- En este apartado se realizará un entrenamiento Baum-Welch de los parámetros de una cadena de Markov Oculta. Para ello, utilizando el modelo `hmm4` genere una secuencia de observaciones X que servirá como datos de entrenamiento. Entre los programas suministrados figura `logfwd.m` que calcula (el logaritmo de) las probabilidades $\alpha_k(t)$, $\forall k, \forall t$, devolviendo la probabilidad final de la secuencia basado en este cálculo. Se debe generar un programa que genere las matrices de valores de $\alpha_k(t)$, $\beta_k(t)$, $\gamma_k(t)$ y $\xi_{jk}(t)$ para toda la secuencia de observación. Utilice el programa mencionado como base para el cálculo de las demás probabilidades. Preste especial atención a la forma en que son tratadas las sumas de probabilidades en dicho programa.
- Utilizando como base el programa de estimación EM generado en la práctica 6, realice las modificaciones necesarias para la estimación de los parámetros de `hmm4`, utilizando las matrices calculadas en punto anterior. Compare los resultados obtenidos con los verdaderos parámetros. Realice dos entrenamientos, uno inicializando todas las medias y varianzas a un mismo valor igual a la media y varianzas globales, y otro con segmentación temporal uniforme.

- c) Repita el entrenamiento usando una secuencia generada por `hmm6`, con ambas inicializaciones.
- d) A partir de los modelos `hmm4` y `hmm6` genere un nuevo modelo de Markov que sea la concatenación de ambos, conectando el estado 4 de `hmm4` directamente al estado 2 de `hmm6`. Esto creará una cadena de 6 estados. Repita el entrenamiento de esta cadena más larga a partir de una observación generada por la misma.

3. Clasificación de secuencias por Máxima Probabilidad

- a) Entre los datos que contiene el archivo `data.mat` se encuentran varias secuencias de características (X) y sus correspondientes secuencias de estados (ST). Utilizando `plotseq2.m`, graficar 6 veces cada secuencia con los 6 modelos posibles como tercer parámetro. Determine intuitivamente a cuál de los modelos corresponde cada salida por medio del gráfico y el conocimiento de la matriz de transición.
- b) Calcule para cada secuencia de características las 6 verosimilitudes que se obtienen para cada uno de los 6 modelos. Luego determine cuál de todos los modelos tiene máxima verosimilitud.
- c) Compare sus hipótesis intuitivas del primer ítem con las obtenidas en el punto anterior.

4. Algoritmo de Viterbi.

- a) La función `logvit.m` calcula el camino óptimo de Viterbi para una dada secuencia obtenida con un modelo de Markov. Encuentre el camino óptimo para cada una de las secuencias del ejercicio anterior, utilizando el modelo que da máxima verosimilitud.
- b) La función anterior calcula también la probabilidad de la secuencia sobre ese camino. Compare dicha probabilidad con la probabilidad de la secuencia sobre todos los caminos posibles (hallada en el ejercicio anterior).
- c) Utilizando la función `compseq.m` compare la secuencia hallada con Viterbi contra la verdadera secuencia.

5. Detección de Modelos Conectados.

- a) Construya un modelo oculto de Markov que pueda generar indistintamente un modelo `hmm4` o `hmm6` con igual probabilidad de empezar en uno o en otro, o de terminar después de haber generado uno o el otro. Debe notarse que este modelo tendrá las mismas medias y varianzas que el modelo construido en el punto 2d, pero la matriz de transición debe ser modificada para que el modelo contemple lo pedido en este punto. Con dicho modelo genere una secuencia de observaciones, y determine la secuencia de estados más probable utilizando `logvit.m`. Compare ambas secuencias. Determine qué *secuencia de modelos se produjo* en dicha emisión.
- b) Repita el reconocimiento de la secuencia de estados con la secuencia generada en el punto 2d