Assignment 3 – Hidden Markov Model

Li-Hen Chen 928003907

Department of Computer Science & Engineering, Texas A&M University

Abstract – Hidden Markov Model is well suited to the task as they involve inference on "hidden" generative processes via "noisy" indirect observations correlated to these processes. In this instance the hidden, or latent process is the underlying regime state, while the asset returns are the indirect noisy observations that are influenced by these states. As a result, HMM is used on the sequential data a lot where sketch can be viewed as one of them. We'll discuss how HMM is applied on the sketch recognition.

a. Discuss the intuition of HMMs and how they can be applied to sketch recognition, plus one or two other domains.

Hidden Markov Model (HMM) is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobservable states. HMM can be defined as below:

Let X_n and Y_n be discrete-time stochastic processes and $n \geq 1$. The pair (X_n, Y_n) is a hidden markov model if

- X_n is a Markov process and is not directly observable ("hidden").
- $P(Y_n \in A | X_1 = x_1, \dots, X_n = x_n) = P(Y_n \in A | X_n = x_n)$,

for every $n \geq 1, x_1, \dots, x_n$, and arbitary measurable set A

The three classical problems associated with HMMs include:

1. Classification Problem: Given an observation sequence, which model is the most likely to have emitted the particular sequence?
2. Most Likely State Sequence Problem: What is the most likely sequence of hidden states which are believed to have emitted a given observation sequence?
3. Training or Learning Problem: Given one or more observation sequences, what are the model parameters associated which maximizing the likelihood of the observations?

Beside sketch recognition HMM can be used speech recognition, musical score following areas.

- For sketch recognition

  When HMM is applied on sketch recognition, it has four main steps including pre-processing, feature generation and pattern recognition.

  1. Pre-processing includes linear and non-linear methods to resample and interpolate a symbol. Factors such as the amount of symbol variation, effects of pixelization, pixel sampling rate and speed which the user draws a symbol result in a possible introduction or elimination of features through pre-processing.

  2. The feature extraction procedure we adopted was a local method of computing an angle from three consecutive pixels. Two vectors are computed, one from the past to present pixel and the other from the present to next pixel. The feature reflects the angle between these two vectors. However, this method has a side effect which is sensitive to the style, speed, and other factors that make discrimination and classification very difficult.

  3. Once we have the appropriate features, we can use these features to train our Hidden Markov Model. One could use the Viterbi algorithm to find the most likely sequence of HMM states and detect points at which the model transitions between states.

- For speech recognition

  We can represent speech as a sequence of observations. Use HMM to model some unit of speech (phone, word) and then concatenate units into larger units. Finally train the model by Forward-Backward Algorithm

- For musical score

  As the same idea of speech recognition, we can represent speech as a sequence of observations. Use HMM to model some unit of note (frequency) and then concatenate units into larger units. Finally also train the model by Forward-

Backward Algorithm.

b. What are some pros and cons of HMMs?

Pros

- The HMM is a well studied probabilistic graphic model, for which algorithms are known for exact and approximate learning and inference
- HMMs are able to represent the variance of appliances' power demands through probability distributions
- HMMs capture the dependencies between consecutive measurements

Cons

- HMMs represent the behaviour of an appliance using a finite number of static distributions, and therefore fail to represent appliances with a continuously varying power demand
- Due to their Markovian nature, they do not take into account the sequence of states leading into any given state
- Again, due to their Markovian nature, the time spent in a given state is not captured explicitly. However, the hidden semi-Markov model does capture such behaviour
- Features other than the observed power demand are not captured (e.g. time of day). However, the input-output HMM allow such such state durations to be modelled
- Any dependency between appliances cannot be represented. However, the conditional-HMM can capture such dependencies

c. When might using a HMM be a good idea? When might it be a bad idea?

HMM is good for sequential data as I mentioned above. As a result, HMMs are especially known for their application in reinforcement learning and temporal pattern recognition such as speech, handwriting, gesture recognition, part-of-speech tagging, musical score following, partial discharges and bioinformatics. All of the domains I listed have the characteristics which are composed of sequential data. However, HMM is not suitable on spatial data such as images. When it comes to vision domain, HMM might be a bad idea.