Spectral Learning of Mixture of Hidden Markov Models

Authored by:

Paris Smaragdis
Cem Subakan
Johannes Traa

Abstract

In this paper, we propose a learning approach for the Mixture of Hidden Markov Models (MHMM) based on the Method of Moments (MoM). Computational advantages of MoM make MHMM learning amenable for large data sets. It is not possible to directly learn an MHMM with existing learning approaches, mainly due to a permutation ambiguity in the estimation process. We show that it is possible to resolve this ambiguity using the spectral properties of a global transition matrix even in the presence of estimation noise. We demonstrate the validity of our approach on synthetic and real data.

1 Paper Body

Method of Moments (MoM) based algorithms [1, 2, 3] for learning latent variable models have recently become popular in the machine learning community. They provide uniqueness guarantees in parameter estimation and are a computationally lighter alternative compared to more traditional maximum likelihood approaches. The main reason behind the computational advantage is that once the moment expressions are acquired, the rest of the learning work amounts to factorizing a moment matrix whose size is independent of the number of data items. However, it is unclear how to use these algorithms for more complicated models such as Mixture of Hidden Markov Models (MHMM). MHMM [4] is a useful model for clustering sequences, and has various applications [5, 6, 7]. The E-step of the Expectation Maximization (EM) algorithm for an MHMM requires running forwardbackward message passing along the latent state chain for each sequence in the dataset in every EM iteration. For this reason, if the number of sequences in the dataset is large, EM can be computationally prohibitive. In this paper, we propose a learning algorithm based on the method of moments for MHMM. We use the fact that an MHMM can be expressed as an HMM with block diagonal transition matrix. Having made that observation,

we use an existing MoM algorithm to learn the parameters up to a permutation ambiguity. However, this doesn?t recover the parameters of the individual HMMs. We exploit the spectral properties of the global transition matrix to estimate a de-permutation mapping that enables us to recover the parameters of the individual HMMs. We also specify a method that can recover the number of HMMs under several spectral conditions.

2 2.1

Model Definitions Hidden Markov Model

In a Hidden Markov Model (HMM), an observed sequence x = x1:T = {x1 , . . . , xt , . . . , xT } with xt ? RL is generated conditioned on a latent Markov chain r = r1:T = {r1 , . . . , rt , . . . , rT }, with 1

rt ? {1, . . . M }. The HMM is parameterized by an emission matrix O ? RL?M , a transition matrix A ? RM ?M and an initial state distribution ? ? RM . Given the model parameters ? = (O, A, ?), the likelihood of an observation sequence x1:T is defined as follows: p(x1:T —?) =

X

p(x1:T , r1:T —?) =

r1:T

T XY

p(xt —rt , O) p(rt —rt?1 , A)

r1:T t=1

=1¿ M A diag(p(xT —

:, O)) ? ? ? A diag(p(x1 — :, O)) ? =

1¿ M

T Y

! Adiag(O(xt)) ?, (1)

t=1

where 1M ? RM is a column vector of ones, we have switched from index notation to matrix notation in the second line such that summations are embedded in matrix multiplications, and we use the MATLAB colon notation to pick a row/column of a matrix. Note that O(xt) := p(xt — :, O). The model parameters are defined as follows: ? ?(u) = p(r1 = u—r0) = p(r1 = u)

initial latent state distribution

? A(u, v) = p(rt = u—rt?1 = v), t ? 2

latent state transition matrix

? O(:, u) = E[xt —rt = u]

emission matrix

The choice of the observation model p(xt —rt) determines what the columns of O correspond to: ? Gaussian: p(xt —rt = u) = N (xt ; ?u , ? 2)

?

O(:, u) = E[xt —rt = u] = ?u .

? Poisson: p(xt —rt = u) = PO(xt ; ?u)

?

O(:, u) = E[xt —rt = u] = ?u .

? Multinomial: p(xt —rt = u) = Mult(xt ; pu , S)

?

O(:, u) = E[xt —rt = u] = pu .

The first model is a multivariate, isotropic Gaussian with mean ?u ? RL and covariance ? 2 I ? RL?L . The second distribution is Poisson with intensity parameter ?u ? RL . This choice is particularly useful for counts data. The last density is a multinomial distribution with parameter pu ? RL and number of draws S. 2.2

Mixture of HMMs

The Mixture of HMMs (MHMM) is a useful model for clustering sequences where each sequence is modeled by one of K HMMs. It is parameterized by K emission matrices Ok ? RL?M , K transition matrices1 Ak ? RM ?M , and K initial state distributions ?k ? RM as well as a cluster prior probability distribution ? ? RK . Given the model parameters ?1:K = (O1:K , A1:K , ?1:K , ?), the likelihood of an observation sequence xn = {x1,n , x2,n , . . . , xTn ,n } is computed as a convex combination of the likelihood of K HMMs: p(xn —?1:K) =

K X

p(hn = k)p(xn —hn = k, ?k) =

k=1

=

=

K X

?k

Tn X Y r1:Tn ,n t=1

K X

(?k

1¿ J

Tn Y

X

?k

k=1

k=1

k=1

K X

p(xn , rn —hn = k, ?k)

r1:Tn ,n

p(xt,n —rt,n , hn = k, Ok)p(rt,n —rt?1,n , hn = k, Ak) !) Ak diag(Ok (xt,n)) ?k ,

(2)

t=1

where hn ? {1, 2, . . . , K} is the latent cluster indicator, rn = {r1,n , r2,n , . . . , rTn ,n } is the latent state sequence for the observed sequence xn , and Ok (xt,n) is a shorthand for p(xt,n — :, hn = k, Ok). Note that if a sequence is assigned to the k th cluster (hn = k), the corresponding HMM parameters ?k = (Ak , Ok , ?k) are used to generate it. 1 Without loss of generality, the number of hidden states for each HMM is taken to be M to keep the notation uncluttered.

2

3

Spectral Learning for MHMMs

Traditionally, the parameters of an MHMM are learned with the Expectation-Maximization (EM) algorithm. One drawback of EM is that it requires a good initialization. Another issue is its computational requirements. In every iteration, one has to perform forward-backward message passing for every sequence, resulting in a computationally expensive process, especially when dealing with large datasets. The proposed MoM approach avoids the issues associated with EM by leveraging the information in various moments computed from the data. Given these moments, which can be computed efficiently, the computation time of the learning algorithm is independent of the amount of data (number of sequences and their lengths). Our approach is mainly based on the observation that an MHMM can be seen as a single HMM with a block-diagonal transition matrix. We will first establish this proposition and discuss its implications. Then, we will describe the proposed learning algorithm. 3.1

MHMM as an HMM with a special structure

Lemma 1: An MHMM with local parameters ?1:K = (O1:K , A1:K , ?1:K , ?) is an HMM with global parame? A, ? ??), where: ters ?? = (O, ? ? ? ? ?1 ?1 A1 0 . . . 0 ? ?2 ?2 ? ? 0 A2 . . . 0 ? ? = [O1 O2 . . . OK] , A? = ? ? , ?? = ? . ? . (3) O .. ? ? .. ? ? . 0

0 . . . AK

?K ?K

Proof: Consider the MHMM likelihood for a sequence xn : !) (Tn K Y X ¿ Ak diag(Ok (xt)) ?k p(xn —?1:K) = ?k 1M

(4)

t=1

k=1

? ?? ? ?1 ?1 0 0 ? ? ? ?2 ?2 ? ? ? diag([O1 O2 . . . OK] (xt))? ? . ? ? =1¿ MK ? ? ? ? .. ? t=1 0 0 . . . AK ?K ?K ! Tn Y ? t)) ??, =1¿ A? diag(O(x MK ?

Tn Y

? A1 0 . . . ? 0 A2 . . . ? .. ? .

t=1

? t). We conclude that the MHMM and an HMM with paramwhere [O1 O2 . . . OK] (xt) := O(x eters ?? describe equivalent probabilistic models.

We see that the state space of an MHMM consists of K disconnected regimes. For each sequence sampled from the MHMM, the first latent state r1 determines what region the entire latent state sequence lies in. 3.2

Learning an MHMM by learning an HMM

In the previous section, we showed the equivalence between the MHMM and an HMM with a blockdiagonal transition matrix. Therefore, it should be possible to use an HMM learning algorithm such as spectral learning for HMMs [1, 2] to find the parameters of an MHMM. However, the true global parameters ?? are recovered inexactly due to noise : ?? ? ?? and state indexing ambiguity via a ? P , A?P permutation mapping P: ?? ? ??P . Consequently,

the parameters ??P = (O ?P) obtained ,? from the learning algorithm are in the following form: ?P = O ? P ¿ , O

? ¿ A?P = P A P , 3

??P = P ?? ,

(5)

where P is the permutation matrix corresponding to the permutation mapping P. The presence of the permutation is a fundamental nuisance for MHMM learning since it causes parameter mixing between the individual HMMs. The global parameters are permuted such that it becomes impossible to identify individual cluster parameters. A brute force search to find P requires (M K)! trials, which is infeasible for anything but very small M K. Nevertheless, it is possible to e using the spectral properties of the global transition efficiently find a depermutation mapping P ? Our ultimate goal in this section is to undo the effect of P by estimating a P e that makes matrix A. A?P block diagonal despite the presence of the estimation noise .

3.2.1

Spectral properties of the global transition matrix

Lemma 2: Assuming that each of the local transition matrices A1:K has only one eigenvalue which is 1, the global transition matrix A? has K eigenvalues which are 1. Proof: ? ? ?? ?? ??1 ? V1 . . . 0 ?1 . . . 0 V1 . . . 0 V1 ?1 V1?1 . . . 0 ? ? ?? ?? ? ? .. A? = ? ? = ? 0 ... 0 ? ? 0 ... 0 ? ? 0 ... 0 ? , . 0 0 0 0 VK 0 0 ?K 0 0 VK 0 0 VK ?K VK?1 — {z } ? V? ?1 V? ?

where Vk ?k Vk?1 is the eigenvalue decomposition of Ak with Vk as eigenvectors, and ?k as a diagonal matrix with eigenvalues on the diagonal. The eigenvalues of A1:K appear unaltered in the ? and consequently A? has K eigenvalues which are 1. eigenvalue decomposition of A,

Corollary 1:

lim A?e = v?1 1¿ ?K 1¿ ?k 1¿ M , M ... v M ... v

e??

(6)

where v?k = [0¿ . . . vk¿ . . . 0¿]¿ and vk is the stationary distribution of Ak , ?k ? {1, . . . , K}. ? ? 1 0 ... 0 ?0 0 . . . 0? ?1 ?V = v k 1¿ Proof: lim (Vk ?k Vk?1)e = lim Vk ?ek Vk?1 = Vk ? .. M. ? e?? e?? . ? k 0 0 ... 0 The third step follows because there is only one eigenvalue with magnitude 1. Since multiplying A? by itself amounts to multiplying the corresponding diagonal blocks, we have the structure in (6). Note that equation (6) points out that the matrix lime?? A?e consists of K blocks of size M ? M where the k?th block is vk 1¿ M . A straightforward algorithm can now be developed for making A?P block diagonal. Since the eigenvalue decomposition is invariant under permutation, A? and A?P have the same eigenvalues and eigenvectors. As e ? ?, K clusters of columns appear in (A?P)e . Thus, A?P can be made block-diagonal by clustering the columns of (A?P)? . This idea is illustrated in the middle row of Figure 1. Note that, in an actual implementation, one would ? to form use a low-rank reconstruction by zeroing-out the eigenvalues that are not equal to 1 in ? ? P)r (V? P)?1 = (A?P)? , where (? ? P)r ? RM K?M K is a diagonal matrix with only (A?P)r := V? P (? K non-zero entries,

corresponding to the eigenvalues which are 1. This algorithm corresponds to the noiseless case A?P . In practice, the output of the learning algorithm ?P e is A?P and the clear structure in Equation (6) no longer holds in (A) , as e ? ?, as illustrated in the bottom row of Figure 1. We can see that the three-cluster structure no longer holds for large e. Instead, the columns of the transition matrix converge to a global stationary distribution. 3.2.2

Estimating the permutation in the presence of noise

In the general case with noise , we lose the spectral property that the global transition matrix has K eigenvalues which are 1. Consequently, the algorithm described in Section 3.2.1 cannot be 4

e: 1
e: 5
e: 10
e: 20
e: 1
rt+1
e: 5
e: 10
e: 20
rt
rt
rt
rt+1 rt
rt
rt
rt
rt
e: 1
e: 5
e: 10
e: 20
rt
rt
rt
rt
rt+1

Figure 1: (Top left) Block-diagonal transition matrix after e-fold exponentiation. Each block converge to its own stationary distribution. (Top right) Same as above with permutation. (Bottom) Corrupted and permuted transition matrix after exponentiation. The true number K = 3 of HMMs is clear for intermediate values of e, but as e ? ?, the columns of the matrix converge to a global stationary distribution. applied directly to make A?P block diagonal. In practice, the estimated transition matrix has only e one eigenvalue with unit magnitude and lime?? (A?P) converges to a global stationary distribution. e and the number of HMM However, if the noise is sufficiently small, a depermutation mapping P clusters K can be successfully estimated. We now specify the

spectral conditions for this. Definition 1: We denote ?Gk := ?k ?1,k for k ? {1, . . . , K} as the global, noisy eigenvalues with —?Gk — ? —?Gk+1 —, ?k ? {1, . . . , K ? 1}, where ?1,k is the original eigenvalue of the k th cluster with magnitude 1 and ?k is the noise that acts on that eigenvalue (note that ?1 = 1). We denote ?L j,k := ?j,k ?j,k for j ? {2, . . . , M } and k ? {1, . . . , K} as the local, noisy eigenvalues with L —?L j,k — ? —?j+1,k —, ?k ? {1, . . . , K} and ?j ? {1, . . . , M ? 1}, where ?j,k is the original eigenvalue th with the j largest magnitude in the k th cluster, and ?j,k is the noise that acts on that eigenvalue. Definition 2: The low-rank eigendecomposition of the estimated transition matrix A?P is defined as Ar := V ?r V ?1 , where V is a matrix with eigenvectors in the columns and ?r is a diagonal matrix with eigenvalues ?G1:K in the first K entries. Conjecture 1: r ?P —?L 2,k —, then A can be formed using the eigen-decomposition of A . Then, ? with high probability, kAr ? Ar kF ? O(1/ T N), where T N is the total number of observed vectors.

If —?GK — ¿

max

k?{1,...,K}

Justification: kAr ? Ar kF = kAr ? A + A ? Ar kF ?kAr ? AkF + kA ? Ar kF =kA ? Ar kF + kA ? A + Ar?kF ?kA ? Ar kF + kAr?kF + kA ? A kF ? ? ?2KM + O(1/ T N) = O(1/ T N), w.h.p., where A is used for A?P to reduce the notation clutter (and similarly Ar for (A?P)r and so on), we used the triangle inequality for the first and second inequalities and Ar? = V ?r?V ?1 , where ?r? is a diagonal matrix of eigenvalues with the first K diagonal entries equal to zero (complement of ?r). For the last inequality, we used the fact that A ? RM K?M K has entries in the interval [0, 1] and we used the sample complexity result from [1]. The bound specified in [1] is for a mixture model, but since the two models are similar and the estimation procedure is almost identical, we are reusing it. We believe that further analysis of the spectral learning algorithm is out of the scope of this paper, so we leave this proposition as a conjecture.

Conjecture 1 asserts that, if we have enough data we should obtain an estimate Ar close to Ar in the squared error sense. Furthermore, if the following mixing rate condition is satisfied, we will be able to identify the number of clusters K from the data. 5

Spectral Longevity of Eigenvalues

No. of Significant Eigenvalues Spectral Longevity

9 8 7 6

K

0

5 4 3 2 1

10

20

10 8 6 4 2 0

1

2

e

3

4
5
6
7
8
9
Eigenvalue Index

Figure 2: (Left) Number of significant eigenvalues across exponentiations. Longevity L?? K 0 with respect to the eigenvalue index K 0 .

(Right) Spectral

ek denote the k th largest eigenvalue (in decreasing order) of the estimated transiDefinition 3: Let ? P ? tion matrix A . We define the quantity, "P 0 # "P 0 #! ? K K ?1 ? e ? l —e X — ? — ? — l l=1 l=1 L?? K 0 := , (7) PM K ? e ¿ 1 ? ? ? PM K ? e ¿ 1 ? ? 0 0 e=1 l0 =1 —?l — l0 =1 —?l — ? K 0 . The square brackets [.] denote an indicator function which outputs as the spectral longevity of ? 1 if the argument is true and 0 otherwise, and ? is a small number such as machine epsilon. Lemma 3: If —?GK — ¿

max k?{1,...,K}

—?L 2,k — and arg maxK 0

e 0 —2 —? K e 0 ——? e 0 — —? K +1 K ?1

= K, for K 0 ?

{2, 3, . . . , M K ? 1}, then arg maxK 0 L?? K 0 = K. Proof: The first condition ensures that the top K eigenvalues are global eigenvalues. The second condition is about the convergence rates of the two ratios in equation (7). The first indicator function has the following summation inside: PK 0 ?1 ? e PK 0 ? e ? 0e l=1 —?l — + —?K — l=1 —?l — . PM K ? e = PK 0 ?1 ? e K ? 0e ? 0 e PM ?0e 0 0 l0 =1 —?l — l0 =1 —?l — + —?K — + —?K +1 — + l0 =K 0 +2 —?l — The rate at which this term goes to 1 is determined by the spectral gap —?K 0 —/—?K 0 +1 —. The smaller this ratio is, the faster the term (it is non-decreasing w.r.t. e) converges to 1. For the second indicator function inside L?? K 0 , we can do the same analysis and see that the convergence rate is again determined by the gap —?K 0 ?1 —/—?K 0 —. The ratio of the two spectral gaps determines the spectral longevity. Hence, for the K 0 with largest ratio

e 0 —2 —? K e 0 ——? e 0 —, —? K +1 K ?1

we have arg maxK 0 L?? K 0 = K.

Lemma 3 tells us the following. If the estimated transition matrix A?P is not too noisy, we can determine the number of clusters by choosing the value of K 0 such that it maximizes L?? K 0 . This corresponds to exponentiating the sorted eigenvalues in a finite range, and recording the number of non-negligible eigenvalues. This is depicted in Figure 2. 3.3

Proposed Algorithm

In previous sections, we have shown that the permutation caused by the MoM estimation procedure can be undone, and we have proposed a way to estimate the number of clusters K. We summarize the whole procedure in Algorithm 1.

4 4.1

Experiments Effect of noise on depermutation algorithm

We have tested the algorithm?s performance with respect to amount of data. We used the parameters K = 3, M = 4, L = 20, and we have 2 sequences with length T for each cluster. We used a Gaussian observation model with unit observation variance and the columns of the emission matrices O1:K were drawn from zero mean spherical Gaussian with variance 2. Results for 10 uniformly 6

Algorithm 1 Spectral Learning for Mixture of Hidden Markov Models Inputs: x1:N :Sequences, M K : total number of states of global HMM. b b,A b b : MHMM parameters Output: ?b = O 1:K 1:K Method of Moments Parameter Estimation ? P , A?P (O) = HMM MethodofMoments (x1:N , M K) Depermutation Find eigenvalues of A?P

Exponentiate eigenvalues for each discrete value e in a sufficiently large range. b as the eigenvalue with largest longevity. Identify K b reconstruction Ar via eigendecomposition. Compute rank-K b clusters to find a depermutation mapping P e via cluster labels. Cluster the columns of Ar with K P P ? and A? according to P. e Depermute O b ? P and A?P Form ? by choosing corresponding blocks from depermuted O . b Return ?. Euclidean Distance vs Sequence Length Euc. Dist.

2
1
0
10
120
230
340
450
560
670
3
3
780
890
1000
T
3
3
3
3
3
3
3
3

Figure 3: Top row: Euclidean distance vs T . Second row: Noisy input matrix. Third row: Noisy reconstruction Ar . Bottom row: Depermuted matrix,

numbers at the bottom indicate the estimated number of clusters. spaced sequence lengths from 10 to 1000 are shown in Figure 3. On the top row, we plot the total error (from centroid to point) obtained after fitting k-means with true number of HMM clusters. We can see that the correct number of clusters K = 3 as well as the block-diagonal structure of the transition matrix is correctly recovered even in the case where T = 20. 4.2

Amount of data vs accuracy and speed

We have compared clustering accuracies of EM and our approach on data sampled from a Gaussian emission MHMM. Means of each state of each cluster is drawn from a zero mean unit variance Gaussian, and observation covariance is spherical with variance 2. We set L = 20, K = 5, M = 3. We used uniform mixing proportions and uniform initial state distribution. We evaluated the clustering accuracies for 10 uniformly spaced sequence lengths (every sequence has the same length) between 20 and 200, and 10 uniformly spaced number of sequences between 1 and 100 for each cluster. The results are shown in Figure 4. Although EM seems to provide higher accuracy on 7

Accuracy (%) of EM algorithm
7 10 13 15 17 20 22 25
60 100 88 81 100100100100100100
60 80 100100100100 80 80 100100
3
5
7
9 12 14 16 18 20 23
78 80 95 100 98 100100100100 79 100
78 60 80 80 100100 80 100100100100
78 1
4
6
8 11 13 14 16 18 20
20 62 86 100100100 78 100100 80
80 1001001001001001001001 80 100
2
4
6
7
9 11 13 14 16 18
56 80 77 82 81 60 100 88 100100 77
56 60 1001001001001001001001001001
56 1
4
5
6
8 10 11 13 14 16
80 100 71 1001001001001001001001
1

3
5
6
7
8
9 11 12 13
34 80 83 66 79 97 69 100 80 78 100
34 40 1001001001001001001001001001001
34 1
3
4
5
6
7
8
9 10 11
80 82 97 65 61 69 69 88 82 80
60 1001001001001001001001001001001
1
3
3
4
5
5
6
7
7
8
12 20 65 73 76 78 77 77 78 63 86
12 80 1001001001001001001001001001001
12 2
3
3
3
3
4
4
5
5
6
1 20 53 68 58 73 79 76 88 58 78 10 31 73 116 158 200 T
1 60 1001001001001001001001001001001 10 31 73 116 158 200 T
1 2 2 2 3 3 3 3 3 3 3 10 31 73 116 158 200 T
100
75
614 1616 2433 2423 2404 3332 5849 4915 6890

56
573 1056 1418 3074 2030 3603 5137 4247 8719
47
846 1093 1434 1851 3258 2396 4330 4133 3629
56
606 969 1873 1646 1892 1861 2311 3914 3609
33
367 550 1241 1323 1098 1943 2662 4431 3920
19
313 724 703 1301 1477 1683 2457 3761 1875
34
187 370 529 734 970 1106 2020 2597 1879
16
178 296 378 754 662 1040 1335 1664 2046
12
5
138 235 427 290 444 588 791 865 855
1
1
27
78
N/K
5
N/K
100 2
80 80 80 85 80 84 100100100 87
Run time (s) of EM algorithm
Run time (s) of spectral algorithm
100 40 100 80 100100100100 80 100100
N/K
N/K
Accuracy (%) of spectral algorithm 100 40 82 100100100100100 75 100100
56
34
10 31
54
89
73
165 172 229 266 233 216
116 T
158
200

Figure 4: Clustering accuracy and run time results for synthetic data experiments. Table 1: Clustering accuracies for handwritten digit dataset. Algorithm

1v2
1v3

1v4
2v3
2v4
2v5
Spectral EM init. w/ Spectral EM init. at Random
100 100 96
70 99 99
54 100 98
83 96 83
99 100 100
99 100 100

regions where we have less data, spectral algorithm is much faster. Note that, in spectral algorithm we include the time spent in moment computation. We used four restarts for EM, and take the result with highest likelihood, and used an automatic stopping criterion. 4.3

Real data experiment

We ran an experiment on the handwritten character trajectory dataset from the UCI machine learning repository [8]. We formed pairs of characters and compared the clustering results for three algorithms: the proposed spectral learning approach, EM initialized at random, and EM initialized with MoM algorithm as explored in [9]. We take the maximum accuracy of EM over 5 random initializations in the third row. We set the algorithm parameters to K = 2 and M = 4. There are 140 sequences of average length 100 per class. In the original data, L = 3, but to apply MoM learning, we require that M K ¡ L. To achieve this, we transformed the data vectors with a cubic polynomial feature transformation such that L = 10 (this is the same transformation that corresponds to a polynomial kernel). The results from these trials are shown in Table 1. We can see that although spectral learning doesn?t always surpass randomly initialized EM on its own, it does serve as a very good initialization scheme.

5

Conclusions and future work

We have developed a method of moments based algorithm for learning mixture of HMMs. Our experimental results show that our approach is computationally much cheaper than EM, while being comparable in accuracy. Our real data experiment also show that our approach can be used as a good initialization scheme for EM. As future work, it would be interesting to apply the proposed approach on other hierarchical latent variable models. Acknowledgements: We would like to thank Taylan Cemgil, David Forsyth and John Hershey for valuable discussions. This material is based upon work supported by the National Science Foundation under Grant No. 1319708.

2 References

[1] A. Anandkumar, D. Hsu, and S.M. Kakade. A method of moments for mixture models and hidden markov models. In COLT, 2012. [2] A. Anandkumar, R. Ge, D. Hsu, S.M. Kakade, and M. Telgarsky. Tensor decompositions for learning latent variable models. arXiv:1210.7559v2, 2012. 8

[3] Daniel Hsu, Sham M. Kakade, and Tong Zhang. A spectral algorithm for learning hidden markov models a spectral algorithm for learning hidden markov models. Journal of Computer and System Sciences, (1460-1480), 2009. [4] P. Smyth. Clustering sequences with hidden markov models. In Advances in neural information processing systems, 1997. [5] Yuting Qi, J.W. Paisley, and L. Carin. Music analysis using hidden markov mixture models. Signal Processing, IEEE Transactions on, 55(11):5209 ?5224, nov. 2007. [6] A. Jonathan, S. Sclaroff, G. Kollios, and V. Pavlovic. Discovering clusters in motion time-series data. In CVPR, 2003. [7] Tim Oates, Laura Firoiu, and Paul R. Cohen. Clustering time series with hidden markov models and dynamic time warping. In In Proceedings of the IJCAI-99 Workshop on Neural, Symbolic and Reinforcement Learning Methods for Sequence Learning, pages 17?21, 1999. [8] K. Bache and M. Lichman. UCI machine learning repository, 2013. [9] Arun Chaganty and Percy Liang. Spectral experts for estimating mixtures of linear regressions. In International Conference on Machine Learning (ICML), 2013.

9