Stochastic Similarity for Validating Human Control Strategy Models

Michael C. Nechyba and Yangsheng Xu

The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Modeling dynamic human control strategy (HCS), or human skill through learning is becoming an increasingly popular paradigm in many different research areas, such as intelligent vehicle systems, virtual reality, and space robotics. Validating the fidelity of such models requires that we compare the dynamic trajectories generated by the HCS model in the control feedback loop to the original human control data. To this end, we have developed a stochastic similarity measure — based on Hidden Markov Model (HMM) analysis — capable of comparing dynamic, multi-dimensional trajectories. In this paper, we first derive and demonstrate properties of the proposed similarity measure for stochastic systems. We then apply the similarity measure to real-time human driving data by comparing different control strategies for different individuals. Finally, we show that the similarity measure outperforms the more traditional Bayes classifier in correctly grouping driving data from the same individual.

1. Introduction

The main strength of modeling by learning, is that no explicit physical model is required; this also represents its biggest weakness, however, especially when the unmodeled process is (1) dynamic and (2) stochastic in nature, as is the case for human control strategy. For such processes, model errors can feed back on themselves to produce trajectories which are not characteristic of the source process or are even potentially unstable [1]. Yet, most learning approaches today, including feedforward neural networks, utilize some static error measure as a test of convergence for the learning algorithm. While this measure is very useful during training, it offers no guarantees, theoretical or otherwise, about the dynamic behavior of the resulting learned model.

Thus, we have developed a similarity measure, based on Hidden Markov Model analysis, as a first step in validating learned models of human control strategy. In this paper, we derive this *similarity measure*; demonstrate its properties through mathematical proof and simulation of known stochastic systems; evaluate the proposed measure on human control data; and show that it outperforms the more traditional Bayes classifier.

2. Stochastic similarity

2.1 Hidden Markov Models

Rich in mathematical structure, HMMs are trainable statistical models, with two appealing features: (1) no *a priori* assumptions are made about the statistical distribution of the data to be analyzed, and (2) a high degree of sequential structure can be encoded. A Hidden Markov Model consists of a set of n states, interconnected through probabilistic transitions. Although algorithms exist for training HMMs with both discrete and continuous output probability distributions, discrete HMMs are preferred to continuous HMMs in practice, due to their relative computational simplicity and lesser sensitivity to initial random parameter settings. Using discrete HMMs for analysis of real-valued signals requires that continuous-valued data be converted to discrete symbols through pre-processing and vector quantization (see Section 2.4 below). Thus, a discrete HMM is completely defined by the following triplet $\lambda = \{A, B, \pi\}$ [2], where A is the probabilistic $n \times n$ state transition matrix, B is the $L \times n$ output probability matrix with L discrete output symbols $\chi \in \{1, 2, \ldots, L\}$, and π is the n-length initial state probability distribution vector.

For an observation sequence O of discrete symbols, we can locally maximize $P(\lambda|O)$ (i.e.probability of model λ given observation sequence O) using the Baum-Welch Expectation-Maximization (EM) algorithm. We can also evaluate $P(O|\lambda)$.

Two HMMs λ_1 and λ_2 are defined to be *equivalent*,

$$\lambda_1 \sim \lambda_2, \text{ iff. } P(O|\lambda_1) = P(O|\lambda_2), \ \forall O \qquad \text{(Eq. 1)}$$

Note that λ_1 and λ_2 need not be identical to be equivalent. The following two HMMs are, for example, equivalent:

$$\lambda_1 = \left\{ \begin{bmatrix} 1 \end{bmatrix}, \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}^T, \begin{bmatrix} 1 \end{bmatrix} \right\},$$

$$\lambda_2 = \left\{ \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}^T \right\} \qquad \text{(Eq. 2)}$$

2.2 Similarity measure

Let O_i, $i \in \{1, 2, \ldots\}$, denote a distinct observation sequence of discrete symbols with length T_i. Also let $\lambda_j = \{A_j, B_j, \pi_j\}$, $j \in \{1, 2, \ldots\}$, denote a discrete HMM locally optimized (using the Baum-Welch algorithm) to maximize $P(\lambda_j|O_j)$. Similarly let $P(O_i|\lambda_j)$ denote the probability of the observation sequence O_i given the model λ_j, and let,

$$P_{ij} \equiv P(O_i|\lambda_j)^{\frac{1}{T_i}} \qquad \text{(Eq. 3)}$$

denote the probability of the observation sequence O_i given the model λ_j, normalized with respect to T_i. In practice, we calculate P_{ij} as,

$$P_{ij} = 10^{\log P(O_i|\lambda_j)/T_i} \qquad \text{(Eq. 4)}$$

to avoid problems of numerical underflow for long observation sequences.

Using the definition in (Eq. 3), Figure 1 illustrates our overall approach to evaluating similarity between two observation se-

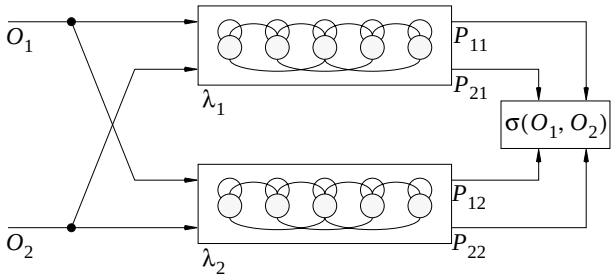

Fig. 1: Four normalized probability values make up the similarity measure.

quences, O_1 and O_2. Each observation sequence first trains a corresponding HMM; this allows us to evaluate P_{11} and P_{22}. Furthermore, we cross-evaluate each observation sequence on the other HMM (i.e. $P(O_1|\lambda_2)$, $P(O_2|\lambda_1)$) to arrive at P_{12} and P_{21}. Given, these four normalized probability values, we now define the following similarity measure between O_1 and O_2:

$$\sigma(O_1, O_2) \equiv \sqrt{\frac{P_{21}P_{12}}{P_{11}P_{22}}} \qquad \text{(Eq. 5)}$$

This measure takes the ratio of the cross probabilities over the training probabilities, and normalizes for the multiplication of the two probability values in the numerator and denominator by taking the square root. We note that in practice, we calculate the P_{ij} not on λ_j itself, but rather $\hat{\lambda}_j$, which is a smoothed version of λ_j, where zero elements in the matrices $\{A_j, B_j, \pi_j\}$ are replaced by $\varepsilon > 0$ and renormalized to fit probabilistic constraints. Throughout this paper, we use $\varepsilon = 0.0001$ as our smoothing value.

2.3 Properties

For now, we assume that P_{ii} is a global (rather than just a local) maximum. Then,

$$1/L \le P_{ii}, \text{ and} \qquad \text{(Eq. 6)}$$

$$0 < P_{ij} \le P_{ii},\ \varepsilon > 0. \qquad \text{(Eq. 7)}$$

The lower bound for P_{ii} in (Eq. 6) is realized for single-state discrete HMMs, and a uniform distribution of symbols in O_i. From (Eq. 5) to (Eq. 7), we can establish the following properties for $\sigma(O_1, O_2)$:

Property #1: $\sigma(O_1, O_2) = \sigma(O_2, O_1)$ (by definition) (Eq. 8)

Property #2: $0 < \sigma(O_1, O_2) \le 1$ (Eq. 9)

Property #3: $\sigma(O_1, O_2) = 1$ iff. $\lambda_1 \sim \lambda_2$ (Eq. 10)

When, P_{ii} is only a local maximum (as guaranteed by Baum-Welch), properties #2 and #3 are only approximately correct. This is not of significant concern, however, as the Baum-Welch algorithm converges to near-optimal solutions in practice.

For the class of single-state, discrete HMMs,

$$\lambda_j = \{A_j, B_j, \pi_j\} = \left\{[1], \begin{bmatrix} b_{j1} & \cdots & b_{jL} \end{bmatrix}^T, [1]\right\}, \qquad \text{(Eq. 11)}$$

which encode only the distribution of symbols without capturing any of the sequential properties of observation sequence O_i, properties #2 and #3 are easy to show. For these HMM's, the similarity measure reduces to,

$$\sigma(O_1, O_2) = \prod_{k=1}^{L} \left(\frac{b_{1k}}{b_{2k}}\right)^{\frac{(b_{2k} - b_{1k})}{2}} \qquad \text{(Eq. 12)}$$

which reaches a maximum when $b_{1k} = b_{2k}$, or simply, $B_1 = B_2$, and that maximum is equal to one. As an example, consider the case where,

$$B_1 = \begin{bmatrix} p_1 & 1 - p_1 \end{bmatrix}^T \qquad B_2 = \begin{bmatrix} p_2 & 1 - p_2 \end{bmatrix}^T \qquad \text{(Eq. 13)}$$

which is graphed in Figure 2 as a contour plot for $0 < p_1, p_2 < 1$.

Finally, we show how the proposed similarity measure changes, as a function of varying HMM structure. Consider the following Hidden Markov Model,

$$\lambda(\alpha) = \left\{ \begin{bmatrix} \frac{1+\alpha}{2} & \frac{1-\alpha}{2} \\ \frac{1-\alpha}{2} & \frac{1+\alpha}{2} \end{bmatrix}, \begin{bmatrix} \frac{1+\alpha}{2} & \frac{1-\alpha}{2} \\ \frac{1-\alpha}{2} & \frac{1+\alpha}{2} \end{bmatrix}, \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} \right\}, \qquad \text{(Eq. 14)}$$

and corresponding observation sequences, $O(\alpha)$, stochastically generated from model $\lambda(\alpha)$. For all $\alpha \in (-1, 1)$, $O(\alpha)$ will have an equivalent aggregate distribution of symbols 0 and 1 — namely 1/2 and 1/2. As $|\alpha|$ increases, however, $O(\alpha)$ will become increasingly structured. For example,

$$\lim_{\alpha \to -1} O(\alpha) = \{\ldots, 1, 0, 1, 0, 1, 0, 1, 0, \ldots\} \qquad \text{(Eq. 15)}$$

$$\lim_{\alpha \to 0} \lambda(\alpha) = \left\{[1], \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}^T, [1]\right\} \qquad \text{(Eq. 16)}$$

$$\lim_{\alpha \to 1} O(\alpha) = \{\ldots, 1, 1, 0, 0, \ldots, 0, 0, 1, 1, \ldots\} \qquad \text{(Eq. 17)}$$

Figure 3 graphs $\sigma[O(\alpha_1), O(\alpha_2)]$ as a contour plot for $-1 < \alpha_1, \alpha_2 < 1$, where each observation sequence $O(\alpha)$ of length $T = 10,000$ is generated stochastically from the corresponding HMM $\lambda(\alpha)$. Greatest similarity occurs for $\alpha_1 = \alpha_2$,

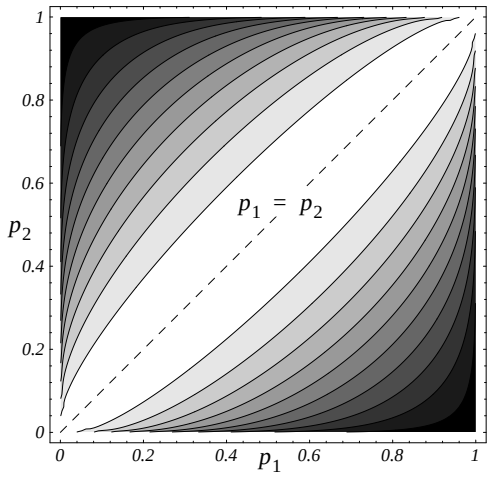

Fig. 2: Similarity measure for two binomial distributions. Lighter colors indicate higher similarity.

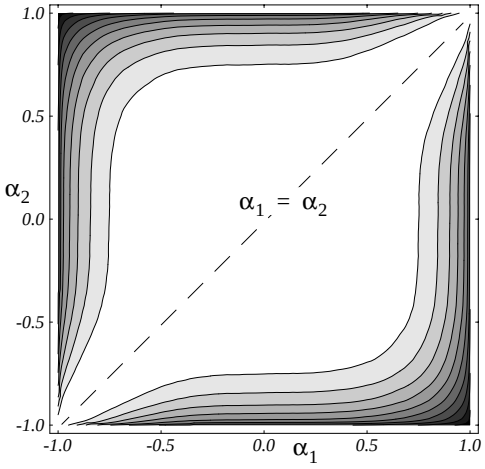

Fig. 3: The similarity measure changes predictably as a function of HMM structure.

and the origin, $(\alpha_1 = 0, \alpha_2 = 0)$, while greatest dissimilarity occurs for $(\alpha_1 \to 1, \alpha_2 = -1)$, and $(\alpha_1 \to -1, \alpha_2 = 1)$.

2.4 Signal-to-symbol conversion

Since we choose to work with discrete-output HMMs, multi-dimensional, real-valued human control data must be converted to a sequence of discrete symbols. Such conversion involves two steps: (1) spectral preprocessing and (2) vector quantization. The primary purpose of the spectral preprocessing is to extract meaningful feature vectors for the vector quantizer. In this work, we rely on the fast Fourier transform (FFT) and the fast Walsh transform (FWT), the $O(n \log n)$ algorithmic counterparts of the discrete Fourier transform (DFT) and the discrete Walsh transform (DWT), respectively. Instead of sinusoidal basis functions, the Walsh transform decomposes a signal based on the orthonormal *Walsh functions* [3]. Certain types of sharply discontinuous human control data are characterized more concisely through the Walsh PSD than the Fourier PSD [4].

For each dimension of the human control data, we partition the data into overlapping window frames, and perform either a short-time FFT or FWT on each frame. Generally, we select the FFT for *state trajectories*, and the FWT for *command trajectories*, since these trajectories tend to have sharp discontinuities for the experimental data in this article. In the case of the FFT, the data in each frame is filtered through a Hamming window before applying the FFT, so as to compensate for the windowing effect. The spectral coefficients are then converted to power spectral density (PSD) vectors. In preparation for the vector quantization, the PSD vectors along each dimension of the system trajectory are normalized and concatenated into one long feature vector per frame. We quantize the resulting sequence of long feature vectors using the iterative LBG VQ algorithm. This vector quantizer generates codebooks of size 2^m, $m \in \{0, 1, 2, \ldots\}$, and can be stopped at an appropriate level of discretization given the amount of available data and complexity of the system trajectories. Assuming that we segment the data into window frames of length k with 50% overlap, the original multi-dimensional, real-valued signal of length t is thus converted to a sequence of discrete symbols of length $T = \text{int}(2t/k)$.

3. Comparing human control strategies

3.1 Experimental set-up

Figure 4 shows the real-time graphic driving simulator which we have developed to collect human control data. The human operator has full control over the steering of the car, the brake and the accelerator. The state of the car is described by [4,5],

$$\{\dot{\theta}, v_\xi, v_\eta\}, \qquad \text{(Eq. 18)}$$

where $\dot{\theta}$ is the angular velocity of the car, v_ξ is the lateral velocity of the car and v_η is the longitudinal velocity of the car; the controls are given by,

$$-8000\text{N} \le P_f \le 4000\text{N}, \qquad \text{(Eq. 19)}$$

$$-0.2\text{rad} \le \delta \le 0.2\text{rad}, \qquad \text{(Eq. 20)}$$

where P_f is the user-applied longitudinal force on the front tires and δ is the user-applied steering angle. Note that the separate brake and gas commands for the human are, in fact, the single P_f variable, where the sign indicates whether the brake or the gas is active. The entire simulator is run at 50 Hz.

3.2 Similarity results

For the first set of experiments, we ask five people — (1) Larry, (2) Curly, (3) Moe, (4) Groucho, and (5) Harpo — to drive on three different, randomly generated 20km roads (similar to the map in Figure 4) for a total of 15 runs. Let,

$$R_{ij} = R(i, j), \; i \in \{1, 2, 3, 4, 5\}, \; j \in \{1, 2, 3\}, \qquad \text{(Eq. 21)}$$

denote the run from person (*i*) on road #*j*. Table 1 reports some aggregate statistics for each of the 15 runs.

Our goal here is to see (1) how well the proposed similarity measure classifies each individual's runs across different roads, and (2) how the classification performance of the proposed similarity measure compares with a more conventional statistical technique, namely the Bayes optimal classifier.

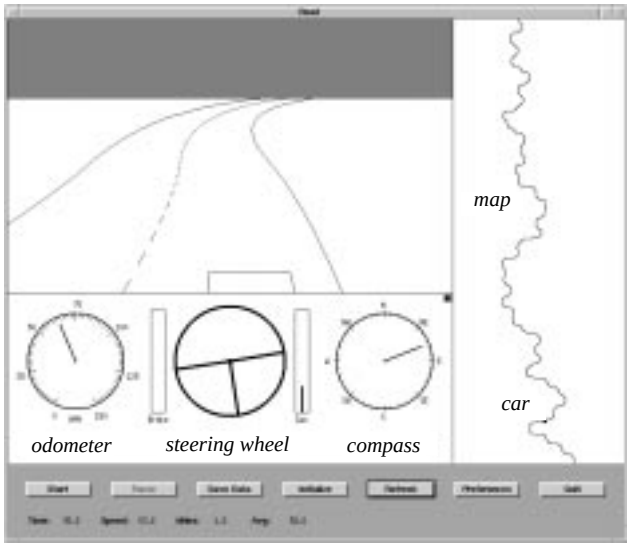

Fig. 4: The driving simulator gives the user a perspective preview of the road ahead. The user has independent controls of the steering, brake, and accelerator (gas).

Table 1: Aggregate statistics for human driving data

	Run	v (mph)	$\dot{\theta}$ (rad/s)	δ (rad)	P_f (1000N)
Larry	*R(1,1)*	63.1 ± 12.2	0.00 ± 0.17	0.00 ± 0.06	1.4 ± 2.4
	R(1,2)	62.7 ± 9.5	0.00 ± 0.17	0.00 ± 0.06	1.3 ± 1.9
	R(1,3)	64.0 ± 8.6	0.00 ± 0.18	0.00 ± 0.06	1.3 ± 1.4
Curly	*R(2,1)*	70.8 ± 8.3	0.00 ± 0.20	0.00 ± 0.07	1.9 ± 3.3
	R(2,2)	69.1 ± 7.7	0.00 ± 0.19	0.00 ± 0.07	1.8 ± 3.3
	R(2,3)	71.5 ± 7.7	0.00 ± 0.20	0.00 ± 0.08	2.0 ± 3.1
Moe	*R(3,1)*	73.1 ± 9.5	0.00 ± 0.24	0.00 ± 0.09	2.2 ± 2.8
	R(3,2)	71.9 ± 9.0	0.00 ± 0.25	0.00 ± 0.09	2.2 ± 2.6
	R(3,3)	74.5 ± 9.4	0.00 ± 0.29	0.00 ± 0.11	2.6 ± 2.6
Groucho	*R(4,1)*	66.8 ± 12.4	0.00 ± 0.18	0.00 ± 0.08	1.9 ± 3.8
	R(4,2)	65.1 ± 13.2	0.00 ± 0.21	0.00 ± 0.09	1.9 ± 4.0
	R(4,3)	69.8 ± 12.3	0.00 ± 0.23	0.00 ± 0.11	2.3 ± 3.8
Harpo	*R(5,1)*	52.3 ± 12.2	0.00 ± 0.17	0.00 ± 0.05	0.9 ± 1.5
	R(5,2)	51.7 ± 4.2	0.00 ± 0.16	0.00 ± 0.04	0.7 ± 0.3
	R(5,3)	56.1 ± 5.7	0.00 ± 0.20	0.00 ± 0.06	1.0 ± 0.3

The system trajectory for the driving task is defined by the three state variables $\{v_\xi, v_\eta, \dot{\theta}\}$ and the two control variables $\{\delta, P_f\}$. We choose the following spectral preprocessing,

$$\{v_\xi, v_\eta, \dot{\theta}, \delta, P_f\} = \{F_{16}, F_{16}, F_{16}, W_{16}, W_{16}\} \quad \text{(Eq. 22)}$$

where F_k, $k \geq 1$, denotes the k-point FFT, and W_k, $k \geq 1$, denotes the k-point Walsh transform. We choose the Walsh transform for $\{\delta, P_f\}$, since these variables are typically step-like in profile.

From the preprocessed data, we build three vector codebooks Q_k, $k \in \{1, 2, 3\}$, each with 128 levels, and corresponding to data from R_{ik}, $\forall i \in \{1, 2, 3, 4, 5\}$. Now define,

$$O_{ij}^k = O(i, j, k),\ i \in \{1, \ldots, 5\},\ j, k \in \{1, 2, 3\} \quad \text{(Eq. 23)}$$

as the observation sequence of discrete symbols vector quantized from the preprocessed feature vectors of run R_{ij}, using the codebook Q_k. The observation sequences O_{ik}^k can be thought of as observation sequences which characterize the control strategy of known individuals, while the observation sequences O_{ij}^k, $j \neq k$, can be thought of as observation sequences which need to be classified. We consider O_{ij}^k, $j \neq k$, classified correctly if and only if,

$$\sigma(O_{ik}^k, O_{ij}^k) > \sigma(O_{lk}^k, O_{ij}^k),\ \forall (j \neq k, l \neq i) \quad \text{(Eq. 24)}$$

In other words, we expect that two runs from the same individual (but on different roads) will yield a higher similarity measure than two runs from two different individuals. Table 2, for example, classifies O_{ij}^k, $j \neq k$, based on O_{lk}^k for k=1 and eight-state HMMs. Note that the maximum value in each row is highlighted, and that the proposed similarity measure correctly classifies all 10 comparisons. Correct classifications are also obtained for k=2 and k=3 [4], but are omitted here due to limited space.

Now we compare these classification results with the Bayes optimal classifier. Define class ω_{ik} as,

$$\omega_{ik} = \omega(i, k) = \{\mu_{ik}, \Sigma_{ik}\}, \quad \text{(Eq. 25)}$$

where μ_{ik} is the mean vector for R_{ik}, and Σ_{ik} is the covariance matrix for run R_{ik}. For each road k, we have five classes, one corresponding to each individual. Each data point $\bar{x} = \{v_\xi, v_\eta, \dot{\theta}, \delta, P_f\}$ in R_{ij}, $j \neq k$, is now classified into class ω_{lk} according to the Bayes decision rule [6],

$$g_{lk}(\bar{x}) > g_{ik}(\bar{x}),\ \forall l \neq i \quad \text{(Eq. 26)}$$

where,

$$g_{ik}(\bar{x}) = -\frac{1}{2}\bar{x}^T\Sigma_{ik}^{-1}\bar{x} + \Sigma_{ik}^{-1}\mu_{ik}\bar{x} - \frac{1}{2}\mu_{ik}^T\Sigma_{ik}^{-1}\mu_{ik} - \frac{1}{2}\log|\Sigma_{ik}| + \log P(\omega_{ik}) \quad \text{(Eq. 27)}$$

Table 3 reports the percentage of data points in R_{ij}, $j \neq k$, which are classified in class ω_{lk} for k=1. We consider R_{ij} to be classified correctly when a plurality of the data from R_{li} falls into ω_{lk}.

Table 2: Similarity measure classification for road #1 data

σ	*O(1, 3, 3)*	*O(2, 3, 3)*	*O(3, 3, 3)*	*O(4, 3, 3)*	*O(5, 3, 3)*
O(1, 1, 3)	0.773	0.144	0.131	0.106	0.376
O(1, 2, 3)	0.597	0.108	0.096	0.053	0.352
O(2, 1, 3)	0.249	0.869	0.340	0.333	0.015
O(2, 2, 3)	0.190	0.803	0.394	0.285	0.009
O(3, 1, 3)	0.280	0.442	0.806	0.397	0.025
O(3, 2, 3)	0.143	0.218	0.680	0.307	0.014
O(4, 1, 3)	0.095	0.210	0.297	0.676	0.010
O(4, 2, 3)	0.065	0.163	0.317	0.679	0.009
O(5, 1, 3)	0.047	0.003	0.005	0.011	0.503
O(5, 2, 3)	0.091	0.009	0.011	0.011	0.614

Table 3: Bayes optimal classification for road #1 data

%	*ω(1, 3)*	*ω(2, 3)*	*ω(3, 3)*	*ω(4, 3)*	*ω(5, 3)*
R(1, 1)	0.458	0.264	0.032	0.096	0.150
R(1, 2)	0.513	0.172	0.014	0.057	0.245
R(2, 1)	0.185	0.575	0.078	0.132	0.030
R(2, 2)	0.208	0.588	0.037	0.129	0.037
R(3, 1)	0.086	0.451	0.270	0.181	0.012
R(3, 2)	0.108	0.424	0.255	0.196	0.018
R(4, 1)	0.201	0.456	0.046	0.292	0.004
R(4, 2)	0.114	0.440	0.083	0.361	0.001
R(5, 1)	0.173	0.010	0.015	0.109	0.693
R(5, 2)	0.059	0.000	0.001	0.000	0.940

In Table 3, for example, 4 out of 10 runs are misclassified. For all the data, the Bayes optimal classifier misclassifies 7 out of 30 (23%) of the runs.

Next, we present results for task-based classification. We select from each run R_{ij} all the left-turn maneuvers, and all the right-turn maneuvers. We get two resulting sets of maneuvers for each person,

$$\alpha_i, \beta_i, \; i \in \{1, 2, 3, 4, 5\} \qquad \text{(Eq. 28)}$$

where α_i corresponds to all the left-turn maneuvers for person *i*, and β_i corresponds to all the right-turn maneuvers for person *i*. We then split each of these sets into two — one to train a VQ codebook (or calculate the Bayesian statistics), the other to determine a similarity value (or the Bayesian classification). Tables 4 and 5 report classification results for the right-turn data. We omit the left-turn tables due to limited space. Once again, the similarity measure classifies all data correctly (both left-turn and right-turn data), while the Bayes classifier misclassifies 30% of the data sets (3 out of 10).

Finally, we present classification results for data which is more difficult to classify. Moe is asked to drive over the *same* road on two different days, two times each day, generating four runs (#1, #2, #3, #4). Because the runs are recorded on the same road, Moe is able to improve his skill relatively quickly. As recorded in Table 6, his average speed improves from 65.9 mph to 71.9 mph from run #1 to run #4. We take two additional data sets, one from Larry and one from Curly, over the same road. These data sets have similar aggregate statistics compared to at least some of Moe's runs. We now wish to classify each of Moe's first three runs as either similar to Larry or Moe #4, or as either similar to Curly or Moe #4.

Tables 7 and 8 show the classification results based on the proposed similarity measure and Bayesian statistics, respectively. We observe that the similarity measure misclassifies one out of six (17%), while the Bayes classifier misclassifies five out of six (83%), some quite badly.

Table 4: Similarity measure classification for right turns

σ	β*(1)*	β*(2)*	β*(3)*	β*(4)*	β*(5)*
β*(1)*	0.713	0.118	0.109	0.036	0.223
β*(2)*	0.119	0.801	0.276	0.199	0.008
β*(3)*	0.109	0.390	0.742	0.324	0.009
β*(4)*	0.032	0.173	0.275	0.773	0.003
β*(5)*	0.244	0.006	0.014	0.005	0.875

Table 5: Bayes optimal classification for right turns

%	β*(1)*	β*(2)*	β*(3)*	β*(4)*	β*(5)*
β*(1)*	0.253	0.188	0.015	0.052	0.492
β*(2)*	0.147	0.616	0.075	0.108	0.054
β*(3)*	0.105	0.330	0.412	0.137	0.015
β*(4)*	0.163	0.364	0.174	0.292	0.008
β*(5)*	0.105	0.014	0.024	0.028	0.830

Table 6: Aggregate statistics for additional human data

Name	*v (mph)*	$\dot{\theta}$ *(rad/s)*	δ *(rad)*	P_f *(1000N)*
Larry	61.9 ± 8.1	0.00 ± 0.18	0.00 ± 0.06	1.3 ± 1.5
Curly	65.6 ± 8.3	0.00 ± 0.20	0.00 ± 0.08	1.8 ± 3.5
Moe #1	66.1 ± 8.2	0.00 ± 0.23	0.00 ± 0.07	1.7 ± 1.8
Moe #2	65.9 ± 8.1	0.00 ± 0.21	0.00 ± 0.07	1.8 ± 2.4
Moe #3	67.4 ± 9.5	0.00 ± 0.23	0.00 ± 0.08	1.9 ± 2.6
Moe #4	71.9 ± 7.4	0.00 ± 0.24	0.00 ± 0.09	2.2 ± 2.1

3.3 Discussion

Figure 5 illustrates why the Bayes classifier fails where the similarity measure succeeds. Figures 5(d), (e), and (f) plot the Gaussian approximation of the distributions (over v and P_f) for Curly's data (Figure 5(a)), Moe's second run (Figure 5(b)), and Moe's fourth run (Figure 5(c)), respectively. It is apparent that the Bayes classifier is doomed to fail, since the human data is distributed in a decidedly non-Gaussian manner. The similarity measure, on the other hand, succeeds because the HMMs are trained on the underlying distributions of the data sets, and make no *a priori* assumptions about each individual's distribution.

In addition, the HMMs are able to encode the sequential properties of the human control data. Figure 6(a) plots the discrimination measure ζ_k,

$$\zeta_k = \frac{\sum_{i,\, l \neq k} \sigma(O_{ik}^k, O_{il}^k)}{\frac{1}{(n-1)} \sum_{i \neq j,\, l \neq k} \sigma(O_{ik}^k, O_{jl}^k)}, \; l, k \in \{1, 2, 3\},$$

$$i, j \in \{1, 2, 3, 4, 5\}, \; n = 5 \qquad \text{(Eq. 29)}$$

(i.e. the ratio of self-similarities over averaged cross-similarities between individuals for observation sequences vector quantized on codebook Q_k) as the number of HMM states is varied from 1 to 8. Figure 6(b) plots a similar ratio for the left-turn/right-turn classifications. From Figure 6, we make two observations: (1) the discrimination of the proposed similarity measure is affected pos-

Table 7: Similarity measure classifications

σ	*Larry*	*Moe #4*	σ	*Curly*	*Moe #4*
Moe #1	0.572	0.528	*Moe #1*	0.315	0.616
Moe #2	0.435	0.540	*Moe #2*	0.495	0.603
Moe #3	0.258	0.728	*Moe #3*	0.550	0.760

Table 8: Bayes optimal classifications

%	*Larry*	*Moe #4*	*%*	*Curly*	*Moe #4*
Moe #1	0.609	0.391	*Moe #1*	0.569	0.431
Moe #2	0.589	0.411	*Moe #2*	0.663	0.337
Moe #3	0.416	0.583	*Moe #3*	0.567	0.433

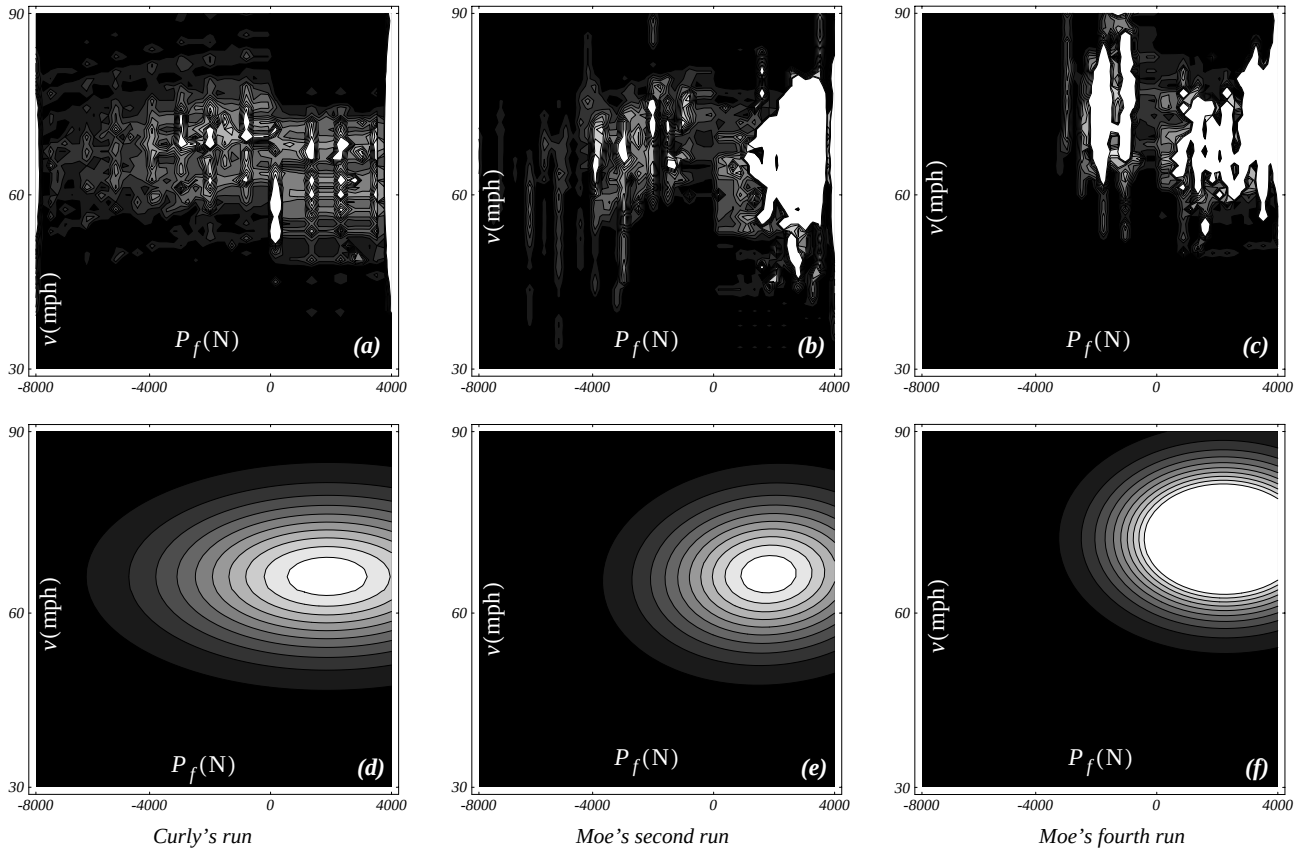

Fig. 5: Distribution over *v* and P_f for (a) Curly's run, (b) Moe's second run and (c) Moe's fourth run; and Gaussian approximation of distribution for (d) Curly's run, (e) Moe's second run and (f) Moe's fourth run.

itively by imparting structure onto the statistical model in the form of an increased number of HMM states, and (2) the discrimination of the similarity measure improves significantly (two-fold in this example), when we train on specific maneuvers (i.e. left turns and right turns) rather than arbitrary roads. Despite various attempts at improving the Bayes classifier's performance, we have yet to identify an example where the Bayes classifier succeeds and the similarity measure fails.

4. Conclusion

In this paper, we have derived a stochastic similarity measure, based on HMM analysis, with which we can compare multi-dimensional stochastic trajectories. We have shown that this method performs significantly better than traditional statistical techniques in classifying human control strategy data. As such, the proposed similarity measure is a key step towards validation of HCS models operating in the control feedback loop.

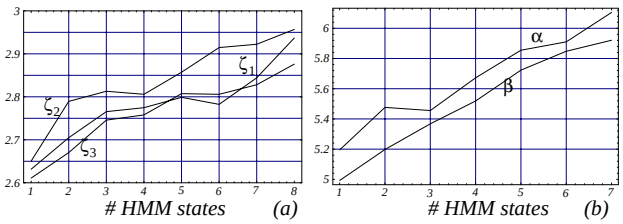

Fig. 6: The discrimination between individuals improves as a function of the number of states in the HMM.

Acknowledgments

We would like to thank the many people who patiently "drove" through the driving simulator. Special thanks go to "Larry", "Curly", "Moe", "Groucho" and "Harpo" for their patience as they navigated through some 50 miles of simulated road.

References

[1] M. C. Nechyba and Y. Xu, "On the Fidelity of Human Skill Models," *Proc. IEEE Int. Conf. on Robotics and Automation*, Vol. 3, pp. 2688-2693, 1996.

[2] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. of the IEEE*, Vol. 77, No. 2, pp. 257-286, 1989.

[3] K. R. Rao and D. F. Elliott, *Fast Transforms: Algorithms, Analyses, Applications*, Academic Press, New York, 1982.

[4] M. C. Nechyba and Y. Xu, "Stochastic Similarity for Validating Human Control Strategy Models," Technical Report, CMU-RI-TR-96-26, Carnegie Mellon University, 1996.

[5] H. Hatwal and E. C. Mikulcik, "Some Inverse Solutions to an Automobile Path-Tracking Problem with Input Control of Steering and Brakes," *Vehicle System Dynamics*, Vol. 15, pp. 61-71, 1986.

[6] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York, 1973.