2-7-2011

Card counting meets hidden Markov models

Steven J. Aragon

Steven Aragon
Candidate

Electrical and Computer Engineering
Department

This thesis is approved, and it is acceptable in quality and form for publication:

Approved by the Thesis Committee:

, Chairperson

Otis M. Solomon, Jr.

CARD COUNTING MEETS HIDDEN MARKOV MODELS

BY

STEVEN J. ARAGON

B.S., ELECTRICAL ENGINEERING
UNIVERSITY OF NEW MEXICO
2001

THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science
Electrical Engineering

The University of New Mexico
Albuquerque, New Mexico

December, 2010

DEDICATION

To my beautiful wife Jean and our soon-to-be-born baby.

ACKNOWLEDGMENTS

Thanks to all my friends and family for helping me along the way. Without your encouragement and prayers, this would not be possible.

Thanks to Professor Ramiro Jordan for not only your expertise, but your patience and willingness to work with me. You were always there to back me up and motivate me.

Finally, thanks to Otis Solomon for dedicating so much time to see me through this. You have been an incredible mentor for many years and I will be forever indebted to you.

CARD COUNTING MEETS HIDDEN MARKOV MODELS

BY

STEVEN J. ARAGON

ABSTRACT OF THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science
Electrical Engineering

The University of New Mexico
Albuquerque, New Mexico

December, 2010

Card Counting Meets Hidden Markov Models

By

Steven J. Aragon

B.S., Electrical Engineering, University of New Mexico, 2001

M.S., Electrical Engineering, University of New Mexico, 2010

ABSTRACT

The Hidden Markov Model (HMM) is a stochastic process that involves an unobservable Markov Chain and an observable output at each state in the chain. Hidden Markov Models are described by three parameters: A, B, and π. A is a matrix that holds the transition probabilities for the unobservable states. B is a matrix that holds the probabilities for the output of an observable event at each unobservable state. Finally, π represents the prior probability of beginning in a particular unobservable state.

Three fundamental questions arise with respect to HMM's. First, given A, B, and π, what is the probability a specific observation sequence will be seen? Second, given A, B, π and an observation sequence, what is the most probable sequence of hidden states that produced the output? Finally, given a set of training data, estimate A, B, and π. There are a number of tools that have been developed to answer these questions.

Woolworth Blackjack is a variation of Blackjack played with a deck consisting of 20 fives and 32 tens. The object is to get a close to 20 as possible without going over.

The player using a basic strategy loses to the dealer. The aim of this research is to develop a winning counting strategy for Woolworth Blackjack and then attempt to improve upon the counting strategy with a HMM using well-established HMM analysis tools. A secondary goal is to understand when to use counting strategies and when to use HMM's.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1. Introduction

A Hidden Markov Model is a finite state automaton consisting of an unobservable Markov process for the transition between states and an observable output for each state. Hidden Markov Models are used extensively in speech recognition, where the goals are to control a device or query an automated system. An example of the former is hands-free operation of mobile devices and an example of the latter is retrieving bank account information over the telephone using voice commands only.

Character recognition in writing is another application of Hidden Markov Models. When writing is processed after generation, it is known as *offline* recognition. Examples of this are automated sorting machines at the Post Office or automatic processing of tax returns at the IRS. There are also *online* recognition methods that exist such as the motion of a pen on a PDA.

In the card game of Blackjack, the object is to get as close to 21 as possible without going over. The suits play no role and each card has a value equal to its rank. Jacks, Queens, and Kings are equal to ten and Aces can be played as one or eleven. Blackjack is generally played in a casino where each player competes against the dealer. Play begins with the dealer and the player each receiving two cards. The dealer leaves one of his cards exposed for the player to see. The player is then allowed to take as many additional cards as he wishes, without going over 21. So long as the player does not go over 21, the dealer plays his hand according to a fixed strategy.

Efficient play of Blackjack rests on employing a basic strategy [8]. A basic strategy is a guide for the player to use based on the cards he is holding and the exposed dealer's card. A basic strategy guides the player to make the "best" decision at any given time. If it does not, it fails to be a basic strategy.

Experienced Blackjack players will often count cards in order to gain an advantage [8, 9, 10]. The idea is that since the dealer plays according to a fixed strategy, there are certain distributions of unused cards in the deck that make it more likely the dealer will go over 21. In practice, this usually means an excess of tens among the unused cards. Thus, card counting is usually not a literal tracking of the number of Aces, twos, threes, etc. played out of the deck, but an assignment of point values to cards or groups of cards. Thus the count is a running sum of the point values after a hand has been played. Depending on the value of the count, the player may change the amount of money he is wagering, violate the basic strategy, or both.

The aim of this research is to study the game of Woolworth Blackjack, which is a variation of Blackjack. The goals are to study the basic strategy as well as determine a card counting strategy, using a point-value system. Ultimately, a deck of cards will be modeled as a Hidden Markov Model and a card counting method based on HMM tools will be developed.

Chapter 2. Markov Chains & Hidden Markov Models

Markov Models are used to model random phenomena such as a system of m distinct states, where the transition from one state to the next is random. If the probability of transitioning from one state to another at time n only depends on the state of the system at time n, then the system is said to be a Markov Model or Markov Chain [1].

As defined in [2], a Markov Chain is a random sequence $X[n]$ whose Nth-order conditional probability mass functions satisfy Equation (2.1) for all n, $x[k]$, and $N > 1$

$$P_X(x[n] \mid x[n-1], x[n-2], ..., x[n-N]) = P_X(x[n] \mid x[n-1]) \qquad (2.1)$$

In the case of discrete-time Markov Chains, $x[n]$ is the state of the system at time n. The state of the system is represented as a positive integer value, i.e. $x = 1,2,\ldots,m$. Therefore, $x[4] = 2$ means that at time $n = 4$ the system is in state 2. It is convenient to drop the equals sign and place the state of the system as a subscript on x. In other words, $x[4] = 2$ is written as $x_2[4]$. If the time is not given or can be inferred by some other context, the brackets are dropped. For a system known to be in state 2, it is simply written as x_2.

A Markov Chain is said to be a finite-state Markov Chain if the range of values $X[n]$ takes on are finite. A Markov Chain is said to be a first-order Markov Chain if the state of the system is only dependent on the previous state of the system [3], as in Equation (2.1).

We define π_i to be the prior probability of being in state x_i at time $n = 0$.

$$\pi_i = P\{x[n] = i\}, n = 0 \tag{2.2}$$

We define the transition probability as the probability of transitioning from state x_i to state x_j by:

$$P\{x_j[n+1] \mid x_i[n]\} = a_{ij} \tag{2.3}$$

The transition probabilities are stored in matrix *A*, which is known as the transition matrix. For a system with c = 4 states

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \tag{2.4}$$

The rows of *A* represent the current state and the columns represent the probability of transitioning to the next state. For model consistency, each row must sum to one: because the a_{ij} in a row are the probabilities of transitioning from state x_i to other states.

Similarly, we write the prior probability, π_i, as a vector:

$$\pi = \begin{bmatrix} \pi_1 & \pi_2 & \pi_3 & \pi_4 \end{bmatrix} \tag{2.5}$$

With *A* and π defined, we have all the information needed to fully describe the Markov Chain. We can then answer questions such as:

- What is the probability of being in any particular state, x_i at a given time?
- What is the probability of transitioning from state x_i to state x_{i+1}?
- For a given sequence of length k, what is the probability the model will follow that particular sequence?

To answer the third question, suppose a sequence of N = 4 state transitions are observed from a state machine that has c = 4 states. The output is as follows

$$X[4] = x_1, x_4, x_3, x_2 \tag{2.6}$$

Since the probability to transition from one state to the next only depends on the current state, we can calculate the probability of the entire sequence, X, as follows:

$$P\{X[4]\} = P\{x_1[0]\} \times P\{x_4[1] \mid x_1[0]\} \times P\{x_3[2] \mid x_4[1]\} \times P\{x_2[3] \mid x_3[2]\} \tag{2.7}$$

$$P\{X[4]\} = \pi_1 \times a_{14} \times a_{43} \times a_{32} \tag{2.8}$$

The Markov chain can be represented by a state-transition diagram, which is a finite-state automaton with edges between any pair of states. The edges are labeled with the transition probabilities, a_{ij}, as seen below in Figure 2.1.

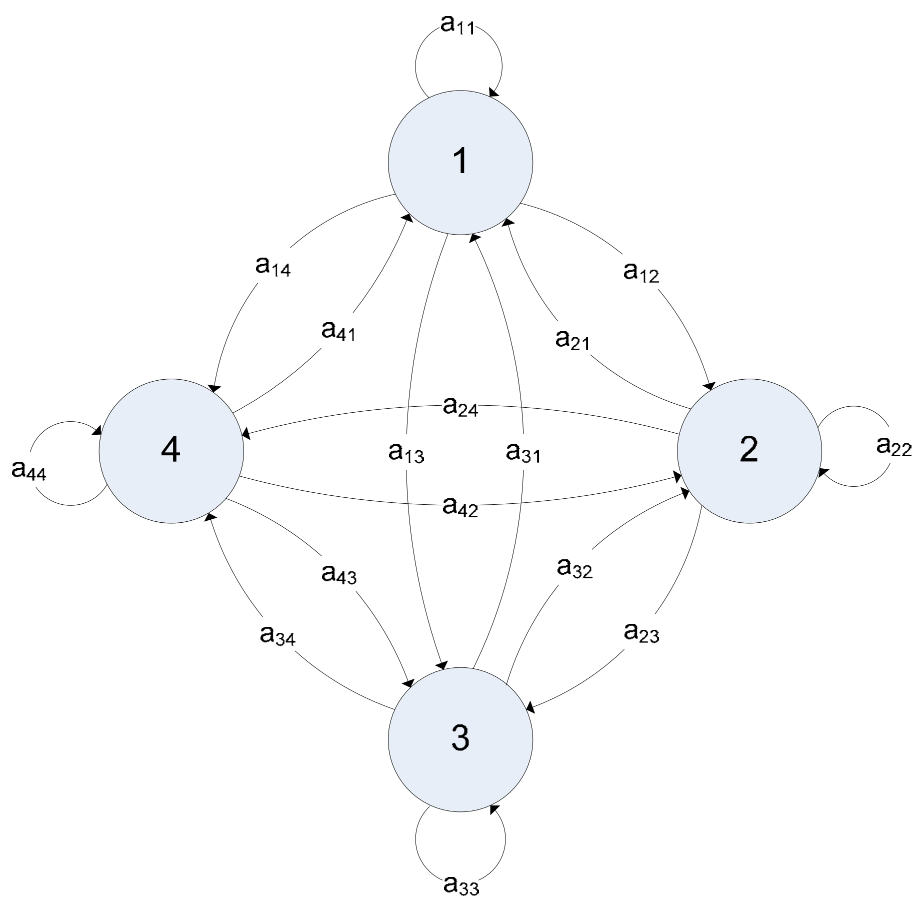

Figure 2.1: 4-state transition diagram

2.1 Hidden Markov Models

The Markov Model is a stochastic process. The transition between states is not deterministic, but follows a probabilistic path. As the finite state machine transitions from state to state, the sequences of states are known as transitions.

Solving problems such as speech recognition require more than just knowledge of the states in a Markov Model. For example [4] if one were to model speech using a Markov Model, then the states might represent phonemes. The word “cat” would have states for /k/, /a/ and /t/ with transitions from /k/ to /a/ and from /a/ to /t/.

However, in speech, one would not be able to “measure” the phonemes, $x[n]$. Rather, we would measure the emitted sound and piece it together with the underlying model to build a word. The emitted sound is known as a visible symbol and is often called an observation. Since the phonemes are not known, they remain hidden and are referred to as hidden states.

More generally, the visible symbols are a second stochastic process that are a probabilistic function of the hidden states. Models such as this are known as Hidden Markov Models (HMM’s). HMM’s are so named because the actual state that produced an observation remains hidden from the observer.

In any state $x[n]$ there is a probability of emitting a visible symbol $v_k[n]$. We define the emission probability as the probability of state x_j emitting symbol $v_k[n]$

$$P\{v_k[n] \mid x_j[n]\} = b_{jk} \tag{2.9}$$

The observation probabilities are stored in matrix B, which is known as the observation matrix. For a system with c = 3 states and k = 4 visible symbols

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \end{bmatrix} \tag{2.10}$$

The rows of B represent the current state of the system the columns show how the probabilities of the visible symbols v_k vary as a function of the state. Each row necessarily sums to one due to the assumption that a visible symbol is always emitted when the machine changes states.

Finally, the prior probability vector, π, completes the specification of the HMM. The prior probability vector is used to determine the initial state of the HMM. The HMM is denoted by θ, where $\theta = (A, B, \pi)$.

There are three fundamental questions that arise with respect to HMM's.

1) Given an HMM, what is the probability that the model generated a particular length n sequences of observations $V[n]=\{v[1],v[2],\ldots,v[n]\}$?
2) Given an HMM and a set of observations, what is the most likely sequence of hidden states that generated the observations?
3) Given a set of training data, estimate the model parameters A, B, and π.

The first question, often referred to as the Evaluation Problem [4], looks at every single length n hidden sequence and determines the probability that it generates the visible sequence. The summation of those probabilities is the probability that the model generated that sequence.

$$P\{V[n]\} = \sum_{r=1}^{R} P\{V[n] \mid w_r\} \times P\{w_r\} \tag{2.11}$$

Where w_1, w_2, w_3,…,w_R are different length n hidden sequences, i.e., w_i is the sequence of states, $\{w[1], w[2],\ldots, w[n]\}$ and $V[n]$ is the sequence of outputs,

$\{v[1], v[2], \ldots, v[n]\}$. If there are c hidden states in the model then there will be $R = c^n$ possible sequences of length n that generated the hidden output sequence.

The second term, $P\{w_r\}$, in Equation (2.11) is recognized as nothing more than the transition probability. As discussed earlier, the first-order Markov process is only dependent on the previous state. Therefore, this term can be written as

$$P\{w_r\} = \prod_{k=1}^{n} P\{w[k] \mid w[k-1]\} \tag{2.12}$$

The first term, $P\{V[n]|w_r\}$, in Equation (2.11) is what was earlier defined as the emission probability. The probability of outputting a given symbol is dependent on the current state of the model. Therefore, this term can be written as

$$P\{V[n] \mid w_r\} = \prod_{k=1}^{n} P\{v[k] \mid w[k]\} \tag{2.13}$$

Finally, Equation (2.11) can be written as

$$P\{V[n]\} = \sum_{r=1}^{R} \prod_{k=1}^{n} P\{v[k] \mid w[k]\} \times P\{w[k] \mid w[k-1]\} \tag{2.14}$$

Recalling the definition of a_{ij} and b_{jk} from above, this can be written as

$$P\{V[n]\} = \sum_{r=1}^{R} \prod_{k=1}^{n} b_{jk} \times a_{ij} \tag{2.15}$$

The literal interpretation of Equation (2.15) is that the probability that a HMM produced a particular sequence is equal to the summation over all R possible hidden state sequences of the conditional probability that a hidden state emitted a specific visible symbol times the probability that a particular state transition was made.

Equation (2.15) is computationally complex $O(nc^n)$ calculation. A far more efficient algorithm, known as the Forward algorithm, has a complexity of $O(nc^2)$. The

Forward algorithm is a recursive algorithm that takes advantage of the fact that at time *k*, Equation (2.15) only depends on $v(k)$, $w(k)$, and $w(k - 1)$.

Given an HMM and an observed sequence $V[n]=\{v[1],v[2],\ldots,v[n]\}$, we define $\alpha_j[t]$ to be the probability that the HMM is in hidden state w_j at time *t*.

$$\alpha_j[t] = \begin{cases} 0 & t = 0 \; and \;\; j \neq initial \; state \\ 1 & t = 0 \; and \;\; j = initial \; state \\ [\sum_i \alpha_i[t-1]a_{ij}]b_{jk}v[t] & otherwise \end{cases} \tag{2.16}$$

We previously defined b_{jk} as the emission probability; the probability of emitting visible symbol v_k given the current state is x_j. The term $b_{jk}v[t]$ refers to the transition probability b_{jk} selected by the visible state emitted at time *t*. Thus, $\alpha_j[t]$ is the probability that the HMM is in hidden state w_j at time *t* given that the first *t* visible symbols of $V[n]$ have been generated.

The Forward algorithm is implemented as follows for an HMM with *c* hidden states [4]:

1. Initialize: $t = 0$, $\alpha_j[0]$, a_{ij}, b_{jk}, $V[n]$
2. For $t = 1$ to n

$$\alpha_j[t] = b_{jk}v[t]\sum_{i=1}^{c}\alpha_i[t-1]a_{ij}$$

3. Return: $P\{V[n]\} = \alpha[n]$

An example of a HMM is the *Urn and Ball Model* discussed by Rabiner [5]. Suppose there are $c = 3$ urns on a table labeled urn 1, urn 2, and urn 3. Each urn has any number of gold, black, and silver balls. The probability of drawing a particular colored ball from a given urn is simply based on the relative frequency. The following events take place, where person A is at the table and person B is in an adjacent room where he can hear person A, but not see what he is doing:

1. Person A selects an initial urn based on the outcome of a roll of a die
2. Person A removes a randomly drawn ball from the urn, announces the color, and returns it to the same urn
3. Person B records the color
4. Person A selects another urn based on the outcome of the roll of a die that is specific to the urn he previously drew from
5. Steps 2, 3 and 4 are repeated *n* times

The example above describes an HMM, where the initial urn is chosen based on the prior probability, π. The visible states or observations are the colored balls, and the hidden states are the actual urns where the balls were drawn from. The probability of drawing a particular color from a given urn is known as the observation probability, b_{jk}. Finally, the probability of transitioning from one urn to the next is known as the transition probability, a_{ij}.

Recall the definition of the transition probability from Equation (2.3). If the probability of transitioning from urn 1 to urn 2 is p_{12}, then

$$P\{Urn_2 \mid Urn_1\} = p_{12} \tag{2.17}$$

Also, recall the definition of the emission probability from Equation (2.9). If the probability of drawing a gold ball from urn 1 is equal to 0.8, it would be written as follows:

$$P\{gold \mid Urn_1\} = b_{Urn_1 gold} = 0.8 \tag{2.18}$$

Therefore, the transition and observation matrices are written as follows:

$$A = \begin{bmatrix} 0 & 2/5 & 3/5 \\ 3/5 & 0 & 2/5 \\ 1/3 & 2/3 & 0 \end{bmatrix} \tag{2.19}$$

$$B = \begin{bmatrix} 0.8 & 0 & 0.2 \\ 0.2 & 0.8 & 0 \\ 0 & 0.2 & 0.8 \end{bmatrix} \tag{2.20}$$

From A, if the current state of the HMM is urn 1, then there is a 60% chance the next state will be urn 3 and a 40% chance the next state will be urn 2. Based on the main diagonal in A, a new urn will be selected at each step; no urn will be chosen twice in a row ($a_{ii} = 0$). While the HMM is in the current state, it will emit a visible symbol. From b_{jk}, there is an 80% chance of drawing a gold ball and a 20% chance of drawing a silver ball from urn 1. From urn 2, there is 20% chance of drawing a gold ball and an 80% chance of drawing black. From urn 3, there is a 20% chance of drawing black and an 80% chance of drawing silver.

Finally, the prior probability, π, is specified. If we assume that the person drawing the balls is equally likely to start at any of the three urns, then

$$\pi = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix} \tag{2.21}$$

Recall the first of the fundamental questions regarding HMM is this: given θ, what is the probability that the model generated a particular length n sequence of visible states? For example, given the HMM defined above, the sequence {gold, black, silver} is observed. What is the probability that the model emitted such a sequence?

This problem is solved via the Forward algorithm. Rabiner [5] implements the Forward algorithm as follows:

1. Initialization:

$$\alpha_1(i) = \pi_i b_{jk}$$

$$\alpha_1 = \begin{bmatrix} (1/3)\times 0.8 \\ (1/3)\times 0.2 \\ (1/3)\times 0 \end{bmatrix}$$

2. Induction:

$$\alpha_{t+1}(t) = \left[\sum_{i=1}^{c} \alpha_t(i) a_{ij}\right] b_{jk}$$

$$\alpha = \begin{bmatrix} 0.2667 & 0 & 0.0127 \\ 0.0667 & 0.0853 & 0 \\ 0 & 0.0373 & 0.0273 \end{bmatrix}$$

3. Termination:

$$P(V[n] \mid \theta) = \sum_{i=1}^{c} \alpha_T(i)$$

$$P\{V[n] \mid \theta) = \alpha_{13} + \alpha_{23} + \alpha_{33} = 0.04$$

There is a 4% chance that the sequence {gold, black, silver} will occur given the HMM defined above.

The second fundamental question deals with the following: given an observed pattern of colored balls {gold, black, silver}, and a complete specification of an HMM, what is the most likely sequence of hidden states that generated the pattern?

The second fundamental problem is the problem of decoding. That is, to discover the hidden sequence that was most likely to have produced a given observation sequence. One of the most efficient ways to determine the most likely hidden sequence is using the Viterbi algorithm [3]. The Viterbi algorithm is a recursive procedure which determines the probabilities of partially optimal paths for each segment. Given a fully defined

HMM, θ, and a set of observations, V, the Viterbi algorithm can be used to determine the optimal state sequence W. This is done by maximizing $P\{W \mid V, \theta\}$.

From Bayes' rule, $P\{W \mid V, \theta\} = P\{W, V \mid \theta\} / P\{V \mid \theta\}$. Thus the joint probability function, $P\{W, V \mid \theta\}$, can be maximized. Define the maximum probability along the most probably state sequence path of a given observation sequence in state i after time t as $\delta_t(i)$.

$$\delta_t(i) = \max_{w_1, w_2, \ldots, w_{t-1}} P\{w_1 w_2 \ldots w_{t-1}, w_t = X_i, v_1 v_2 \ldots v_t \mid \theta\} \tag{2.22}$$

The Viterbi algorithm is implemented as follows [5]:

Step 1: Initialize

$$\delta_1(i) = \pi_i b_{ik}$$
$$\psi_1(i) = 0$$

Step 2: Recursion

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_{ik}, \quad 2 \le t \le T, \quad 1 \le j \le N$$
$$\psi_t(j) = \arg\max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}], \quad 2 \le t \le T, \quad 1 \le j \le N$$

Step 3: Termination

$$P^* = \max_{1 \le i \le N} [\delta_T(i)]$$
$$w_T^* = \arg\max_{1 \le i \le N} [\delta_T(i)]$$

Step 4: Backtracking

$$w_t^* = \psi_{t+1}(w_{t+1}^*), \quad T - 1 \ge t \ge 1$$

The third fundamental problem with HMM's deals with the following: given a set of training data and a loose definition of an HMM, determine the HMM parameters,

$\theta = (A, B, \pi)$. Often the loose definition of the HMM is the number of hidden and visible states. The general idea is to maximize $P\{V \mid \theta\}$.

One of the most common ways to estimate θ is using the Baum-Welch algorithm. Baum-Welch is an iterative algorithm used to find a local maximum of $P\{V \mid \theta\}$. It first calculates the Forward probability for a state followed by the Backward probability. The iterative nature of Baum-Welch allows a local maximum to be found, but does not guarantee a global maximum.

Chapter 3. Woolworth Blackjack

Woolworth Blackjack [9] is a variation of Blackjack played with a special deck of cards containing 20 fives and 32 tens. The play is similar to Blackjack; players do not compete with each other but play against the dealer. The suits play no role in the game and the cards carry a point value equal to their rank. After placing a wager, the player and the dealer are each dealt two cards. One of the dealer's cards is face up (the "up card") while the other is face down (the "down card").

The player is the first to act, deciding to take on additional cards ("hit") or keep the cards he has ("stand"). He may continue to hit until the sum of the rank of the cards in his hands is greater than twenty (a "bust"). If the player busts, he loses his wager and there is no action taken by the dealer on his own hand. Otherwise, the dealer is forced to act. Regardless of the point total for the player, the dealer must continue to hit until he has a total of twenty or he busts. If the dealer busts, then he pays the player even money on his wager. If the dealer reaches twenty, then the settlement occurs as follows: if the player also has twenty, the hand ends in a tie or a "push" and no money changes hands. Otherwise, if the player has less than twenty, the dealer wins and he takes the player's wager. The player is not allowed to split pairs at any time. He may, however, double down when he has a total of ten. Doubling-down is the process where a player doubles his wager on the first two cards he is dealt and receives exactly one additional card.

After the settlement is complete, the dealer collects all the used cards. He may then put those cards aside and deal the next hand from the remaining unused cards, or he may shuffle the entire deck before dealing the next hand.

The basic strategy for the player is as follows:

1. Stand on a total of 15 or higher against any dealer up card
2. Hit a total of 10 against a dealer up card of 10
3. Double-down 10 against a dealer 5

The expectation for the basic strategy is -0.63% [9].

To verify the expected return of -0.63%, ten independent simulations of 100,000 hands of Woolworth blackjack were played with the basic strategy. Each hand was played with a full deck. All cards were returned to the deck and the deck was reshuffled before the next hand. Based on a wager of one unit, the simulation average return was *Average_gain* = -0.0057 ± 0.0023 units. The expected return of -0.63% falls within one standard deviation of the simulation average.

Generally a player will assume that the basic strategy is the most efficient method of play regardless of the number of cards left in the deck. This assumption will be discussed further in the next section.

3.1 Multiple Hand Play

Suppose instead of playing a single hand from the deck, multiple hands are played without replacement of cards. That is, the cards that have been played are put aside until the dealer reshuffles the entire deck. It was observed during simulation that for a single player playing against the dealer and using the basic strategy, there were sufficient cards to play as many as nine hands 100% of the time. Ten hands could be played 99.96% of the time and there were sufficient cards to play eleven hands 34.1% of the time. Play of twelve hands from a single deck without shuffling was extremely rare, occurring less than three times for every ten million decks.

Towards the end of the deck, specifically after the 9^{th} hand, it will not be obvious if there are sufficient cards to finish a hand. The question of what to do when all cards have been dealt out of the deck but the hand is not over raises a dilemma. In the game of Rummy, if the stock pile is depleted before a player "knocks", the top card from the discard pile is set aside and the remainder of the discard pile is shuffled. The shuffled cards are placed face down and become the new stock pile. The top discard becomes the first card in the new discard pile [11].

For the purposes of analysis, it is desired to maximize the number of Woolworth Blackjack hands played per deck. However, reshuffling of used cards to finish a hand will likely introduce statistics at the end of the deck that are not meaningful. Therefore, if a hand cannot be finished because all the cards are dealt out of the deck, the hand will be considered unplayable. The player's wager will be returned to him and all cards will be collected by the dealer for reshuffling. Additionally, the maximum number of hands dealt will be limited to eleven, since play of twelve hands is so incredibly infrequent.

Given a single player using the basic strategy against the dealer with a maximum of eleven hands per deck, the simulation results yield an average gain of -1.50% for the player. This is much worse than the single hand expectation of -0.63% given by Griffin.

An immediate conclusion to minimize losses would be for the player to only play the first hand from a fresh deck.

Although it may seem simplistic to simply refrain from playing more than one hand per deck, given the disastrous effects of playing multiple hands it does not seem like such a bad idea. However, this strategy leaves much to be desired. It may be that the second hand also has an expectation of -0.63%, and therefore the player is at no more of a disadvantage to play the second hand. Possibly the third hand is profitable for the player. In any event, an average gain for the deck of -1.50% implies that there may be a breakdown in the basic strategy. If so, the first step is to determine where it occurs and the second step is to alter the strategy in order to compensate for the breakdown. An example of this would be if all the tens in the Woolworth deck were played out leaving only fives. Basic strategy guides the player to double down on ten vs. dealer's five, for a total of fifteen. The dealer will draw to twenty and win the hand. In this case, there is a breakdown of the basic strategy when all tens have been played out of the deck. The alteration of the strategy would be for the player to simply draw to twenty for a push.

To determine if there is indeed a breakdown in the basic strategy, consider the average gain per *hand*. The results of simulation, broken down by average gain per hand, are shown in Figure 3.1.

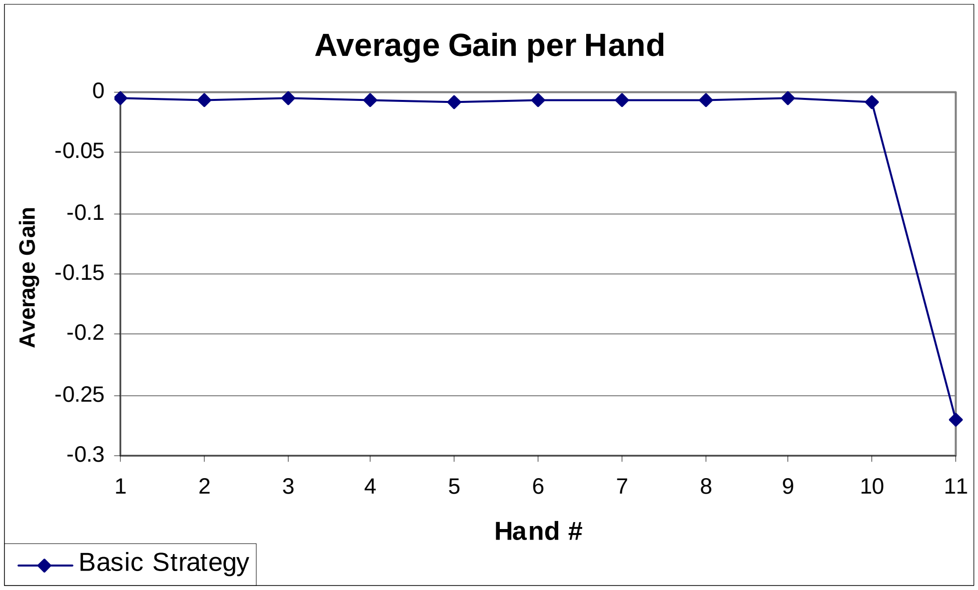

Figure 3.1: Average gain per hand using Basic Strategy

As can be seen in Figure 3.1, the average gain on the 11th hand is considerably worse than any other hand. With an average gain of -27%, there is clearly something happening at the end of the deck causing a breakdown of the basic strategy. The average gain between hand 1 and hand 10 is -0.64% with a standard deviation of 0.11%. The minimum value is -0.82% on hand 10 while the maximum is -0.47% on hand 3.

For the sake of simplicity, the 11th hand will simply not be played. Results of simulations imply that the basic strategy holds for the first ten hands. It is beyond the scope of this research to study what occurs at the end of the deck.

3.2 Card Counting

One way for a player to improve the average gain would be for him to recognize what conditions exist that make it profitable to deviate from the basic strategy and play accordingly. A second way would be for the player to recognize the conditions that make it profitable for him to change his betting structure. A third way would be some combination of the first two. However, for a player to deviate from basic strategy or even change his betting structure, he would have to have knowledge of the distribution of the cards remaining in the deck. This implies a count, or a tracking of those cards already played. One approach to developing a counting strategy is to begin by determining the effects of removal of cards from a complete deck [8, 9, 10].

To study the effects of removal of cards from the deck, Griffin [9] claims the single hand expectation, using basic strategy, for a deck with one five removed is -0.01%. The single hand expectation for a deck with one ten removed is -1.02%. These numbers yield improvements of +0.62% and -0.39%, respectively. The immediate conclusion one may draw is that the removal of fives is beneficial for the basic strategy while removal of tens is detrimental. Results of ten independent simulations to verify Griffin's claims are given in Figure 3.2:

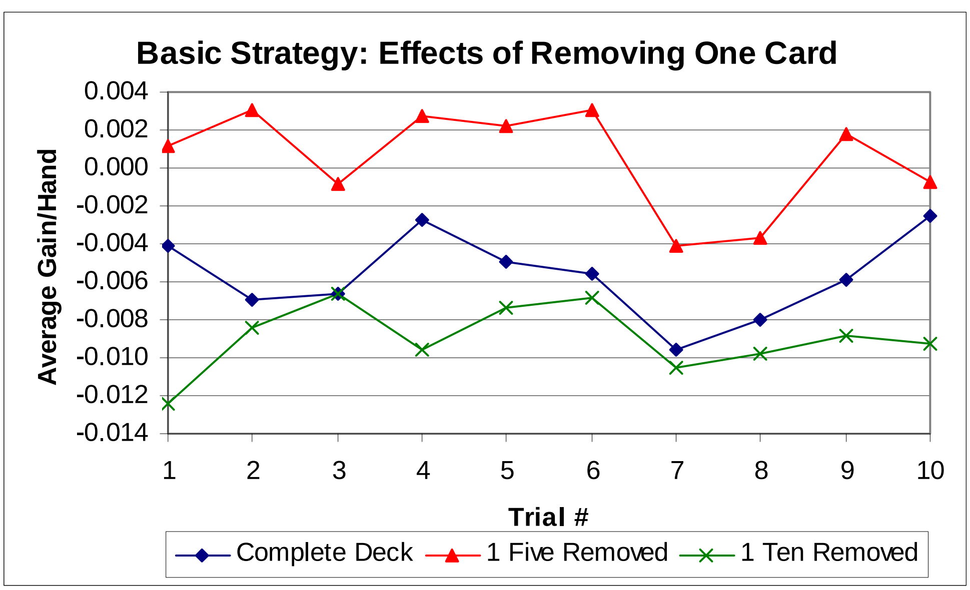

Figure 3.2: Effects of removing a single card on a single hand of Woolworth blackjack.

The simulations shown in Figure 3.2 reflect what is expected to happen when a single five or a single ten are removed from the deck compared to a complete deck. Removal of a five yielded an improvement while removal of a ten yielded an even greater loss. Recall that the expectations for the complete deck, the deck with one five removed, and the deck with one ten removed are -0.63%, -0.01%, and -1.02%, respectively.

Since removing a single five from the deck improved the basic strategy by +0.62%, one is inclined to further study the effects of removal of n fives. On the other hand, a study of the removal of k tens is also warranted.

Simulations of single hands with basic strategy were performed by holding the number of tens constant at 32 and removing from 1 to 19 fives from the deck (removing twenty fives would yield a deck entirely composed of tens and every hand would be a push). Simulations were also performed by holding the number of fives constant at 20 and removing from 1 to 31 tens from the deck (removing 32 tens would yield a deck entirely composed of fives and the player would lose every hand). Results are given in Figures 3.3 and 3.4.

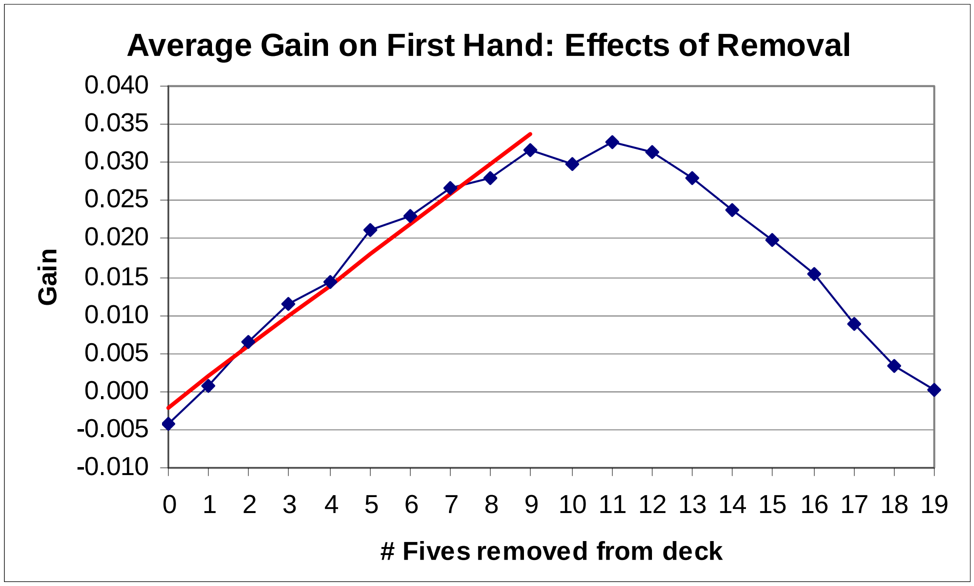

Figure 3.3: Effects of removing multiple fives on a single hand of Woolworth blackjack.

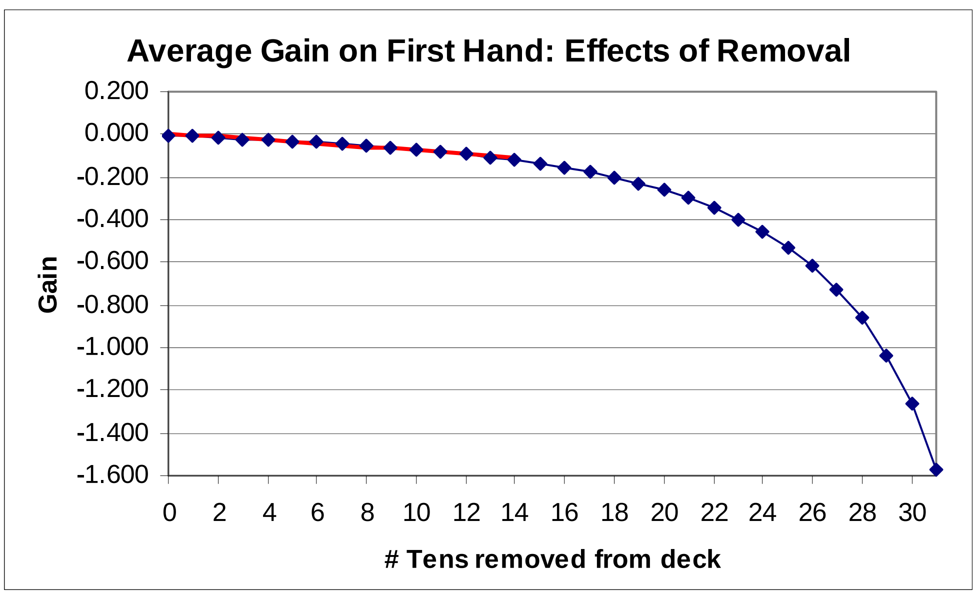

Figure 3.4: Effects of removing multiple tens on a single hand of Woolworth blackjack.

From Figure 3.3, the removal of any number of fives is clearly better than the removal of no fives. Additionally, the removal of *n* fives is advantageous for a single hand. Although the advantage peaks and eventually falls, it is in the player's interest to note that the removal of fives is advantageous. On the other hand, as can be seen in Figure 3.4, removal of tens is bad for the player, no matter how many tens are removed. In fact, once about 18 or so tens have been removed, the expected gain falls off like $-x^2$.

Figure 3.3 has a linear range for about the first ten data points (0 to 9 fives removed) and a linear fit, indicated in red, is included on the chart. The slope of this line is $m_5 = 0.004$. Given that the Woolworth deck has 20 fives and 32 tens, there are 1.6 tens for every five. Therefore, when 9 fives have been removed from the deck, 14.4 tens will have been removed. A linear fit for the first 15 data points (0 to 14 tens removed) is included in Figure 3.4 and is indicated by the red line. The slope of this line is $m_{10} = -0.008$.

The ratio of the slopes, m_{10} to m_5 is equal to -2. By assigning a value of +2 to fives and -1 to tens, a counting system based on these point values can be defined. The Counting Strategy is implemented as follows: when the deck has been shuffled, the count is zero. For every ten that is seen, subtract 1 and for every five that is seen, add 2 to the count. Bet when the count is positive, that is, when more fives have been removed from the deck than tens. Employ the basic strategy when playing. Results of simulation are as follows:

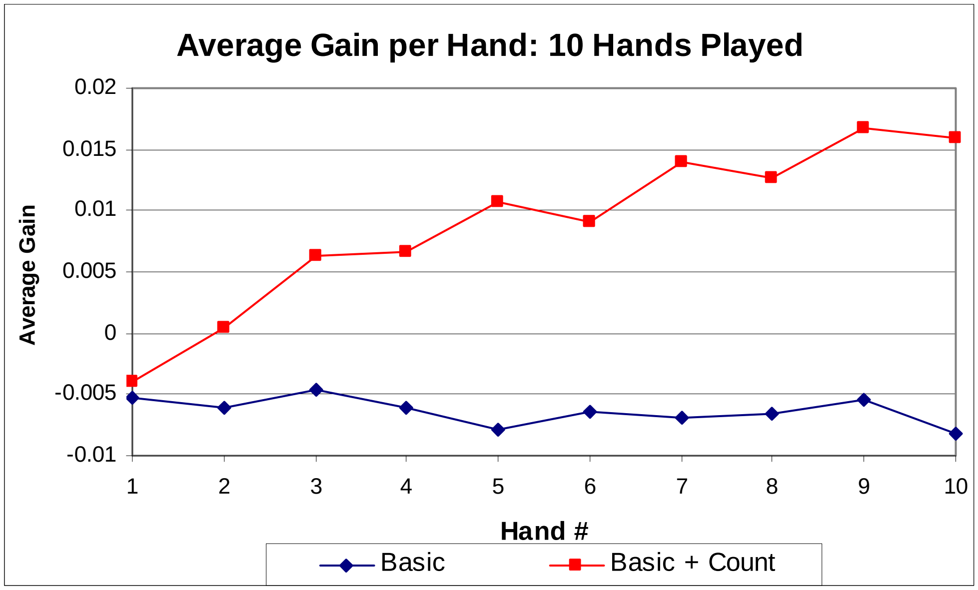

Figure 3.5: Average return per hand; up to 10 hands played.

With the number of hands played limited to 10, Figure 3.5 shows the improvement in the counting strategy over the basic strategy. For all but the first hand, the average gain is positive, reaching a maximum of about 1.68% on the ninth hand. For the entire deck, the counting strategy has an overall gain of 0.903% while the basic strategy has an overall gain of -0.635%. Recall the expected return for a single hand using basic strategy is -0.63%. The results of simulation suggest that the expectation of -0.63% holds not only for the *first* hand, but for *any* hand up to the 10th hand.

It should be noted at this point that the structure of the game rests on the assumption that the player, regardless of the strategy employed, places a wager on the first hand. This is why the return on the first hand using the basic strategy has the same average gain as the counting strategy in Figure 3.5. On the first hand, no cards have been turned over, and thus there is no additional information for the counting strategy to use.

It is also worth noting that the counting strategy employed here has a bias associated with it. Were the player to count the entire deck, he would end up at +8 after the 52nd card. This bias could lead to the player betting in unfavorable situations or refraining from betting in favorable situations.

To eliminate this bias, recall that there are 1.6 tens in the deck for every five. Based on the data from Figures 3.3 and 3.4, an assumption was made that removal of any number of fives was beneficial for the player, while removal of tens was detrimental. Therefore, an *unbiased count* can be developed by setting the value of fives equal to +1.6 and keeping the value of tens equal to -1. This will ensure the count of the deck will begin and end at zero. Comparison of the average gain per hand for the basic strategy, count strategy, and unbiased count strategy are given in Figure 3.6.

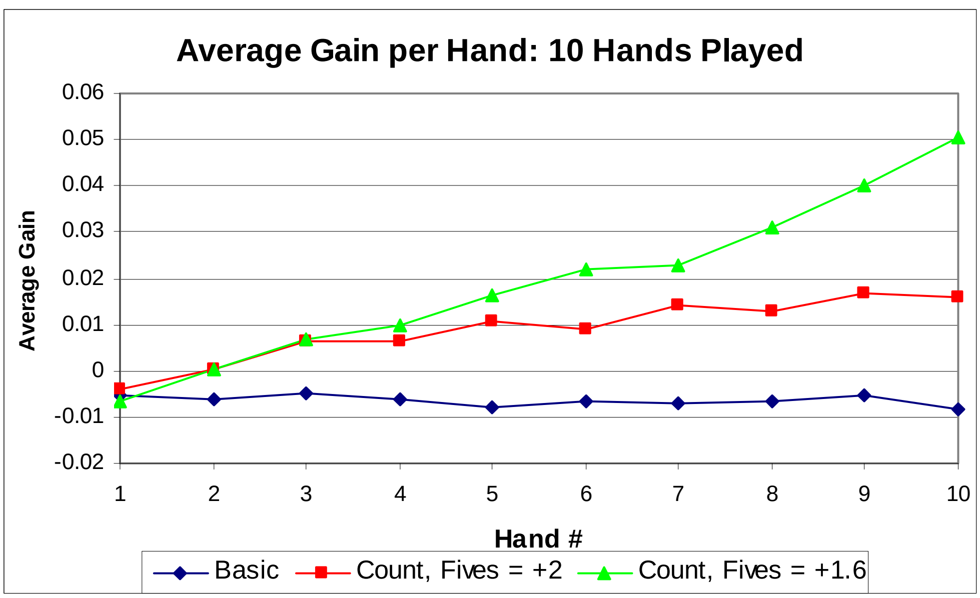

Figure 3.6: Comparison of counting methods for ten hands

From Figure 3.6, it can be seen that the unbiased count improves the player's gain relative to the biased count after the third hand. When only a few cards have been dealt, the negative effects of the bias do not show up. It is not until later in the deck that the bias indicates favorable situations where they do not exist. There is a remarkable improvement using the unbiased count: the gain after ten hands is 1.658%. Recall the gain for the basic strategy is -0.635% and the gain using the biased count is +0.903%.

For the sake of simplicity, a player may wish to use the biased count even though the rate of return is lower. It is easier to add whole numbers (2's) than decimal numbers (1.6's), particularly when playing against a fast-moving dealer. Additionally, by assigning a value of +2 to the fives, cards can quickly be grouped together (i.e. two fives and one ten cancel each other out).

However, for the remainder of this paper, the unbiased count will be used and will simply be referred to as the counting strategy.

3.3 Defining States

The first step in modeling a deck of cards as a Hidden Markov Model (HMM) is defining the model parameters: the Hidden States, the Observations, and the Prior probability vector. One way to do this would be to define the number of fives and tens remaining in the deck as the hidden states and the cards turned over as the observations. The prior would be the deck after shuffling. In this case, if the initial state is the full deck, then the total number of fives is 20 and the total number of tens is 32. Denote this as (# of fives, # of tens), or in this case, (20, 32). If one five is removed from the deck, state (20, 32) transitions to state (19, 32) with probability 20/52. If one ten is removed from the deck, state (20, 32) transitions to state (20, 31) with probability 32/52. Defining the states in such a manner would result in 693 total states. An abbreviated state transition diagram, in the form of a tree, illustrates this model in Figure 3.7.

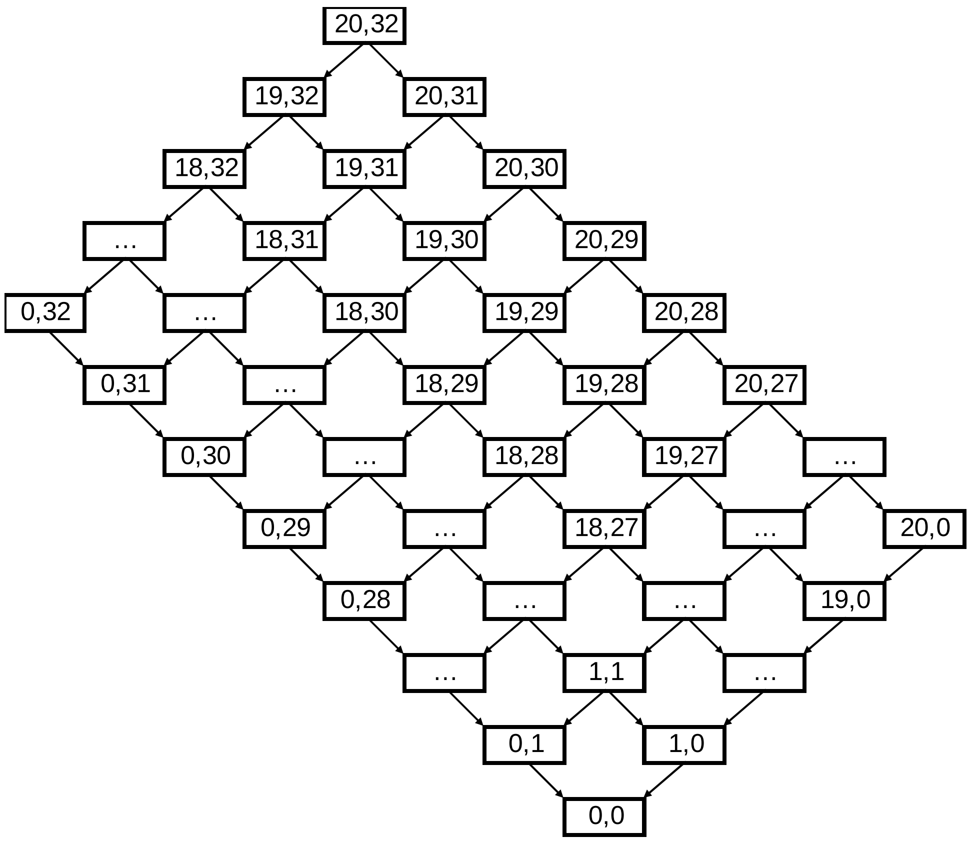

Figure 3.7: Abbreviated Tree diagram representing distribution of cards in a Woolworth deck.

The abbreviated tree diagram in Figure 3.7 represents 693 possible states for a Woolworth deck. Movement to the left represents removal of fives while movement to the right represents removal of tens. The transition probability from a given state (*f*, *t*) to one of the children can be computed by dividing the target card by the sum of the number of total cards. Therefore state (*f*, *t*) transitions to state (*f* – 1, *t*) with probability

$$a_{ij} = P\{(f-1,t)\,|\,(f,t)\} = \frac{f}{f+t} \tag{3.1}$$

The prior probability vector takes on the following form, where π_1 = (20, 32)

$$\pi = \begin{bmatrix} 1 & 0 & 0 & \ldots & 0 \end{bmatrix} \tag{3.2}$$

The vector π is 1x693 since there are a total of 693 possible distributions of a deck containing *f* fives and *t* tens where *f* takes on any integer value from 0 to 20 and *t* takes on any integer value from 0 to 32. The complete deck has 20 fives and 32 tens and therefore the HMM will always begin with a full deck, or π_1.

The naming of states in an HMM is essentially arbitrary; each state can be assigned a number, a letter, or even a name. Thus far, the states have been referred to as the number of fives, *f*, and the number of tens, *t*, in the deck in the form of (*f*, *t*). Figure 3.8 represents an abbreviated tree diagram where (*f*, *t*) has been replaced by a number. State 1 has already been defined as (20, 32).

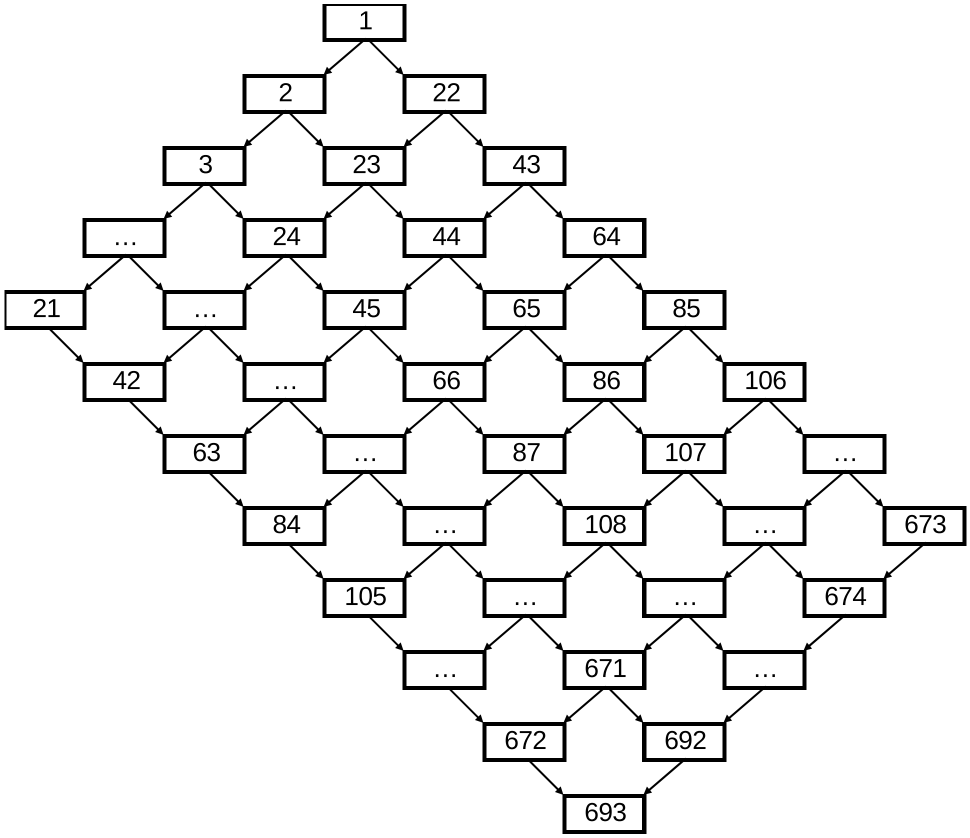

Figure 3.8: Abbreviated Tree diagram representing State numbers

The basic strategy expected gain of -0.63% implies that certain distributions of cards are favorable to the player, while other distributions are favorable to the dealer. The negative expectation implies that there are more of the latter. Use of the counting strategy allows the player to compensate for distributions that are less favorable and exploit distributions that are more favorable. As was illustrated in section 3.2, playing one hand with a single five removed increases the players expectation by 0.62% while playing one hand with a single ten removed decreases the expectation by 0.39%. These numbers were verified via simulation and are shown in Figure 3.2.

The counting strategy was born based on single hand play by removing *f* fives from the deck while holding the number of tens constant and then by removing *t* tens from the deck while holding the number of fives constant. Results of those simulations were given in Figures 3.3 and 3.4. However, removal of *f* fives and *t* tens at the same time was not explored. To do such a thing, one would have to simulate play from each of the 693 states in Figure 3.8, less the states that are unplayable, as defined in section 3.1 (i.e. state 671, where the deck consists of one five and one ten).

Consider that outside of the unplayable hands, there are some specific card combinations where a hand may or may not be completed, depending on the ordering of the cards. For example, a deck consisting of 3 fives and 2 tens (state 648) cannot always be finished. If the player has (F, F) vs. dealer (T, T), the basic strategy will guide the player to hit his total of ten against the dealer's ten. The hand will end in a loss of player (F, F, F) vs. dealer (T, T). On the other hand, if the player is dealt (T, T), then the dealer does not have enough cards to finish his hand (F, F, F). Therefore, a deck is said to be exhausted when a hand cannot be completed 100% of the time. The unplayable hands are a subset of the exhausted hands and can be seen in Table 3.1.

Table 3.1: The exhausted hands

# Fives	# Tens
6	0
5	1
5	0
4	1
4	0
3	2
3	1
3	0
2	2
2	1
2	0
1	3
1	2
1	1
1	0
0	3
0	2
0	1
0	0

Performing a complete removal analysis of the full 52 card deck would require repeated trials of basic strategy play for each state as defined in Figure 3.8, less the exhausted states, outlined in Table 3.1. The gain for the exhausted states is taken to be zero, i.e. nothing happens to the player's wager when the deck has been exhausted. Additionally, it is not necessary to simulate states where either the fives or the tens have been completely removed from the deck. The results are a push in the former case and a loss of 2 (double-down bet) in the latter case. This leads to a reduction from 693 states to 631 states. Figure 3.9 shows the average gain per state after simulation of the 631 states, with the known gain for the remaining 62 states also plotted.

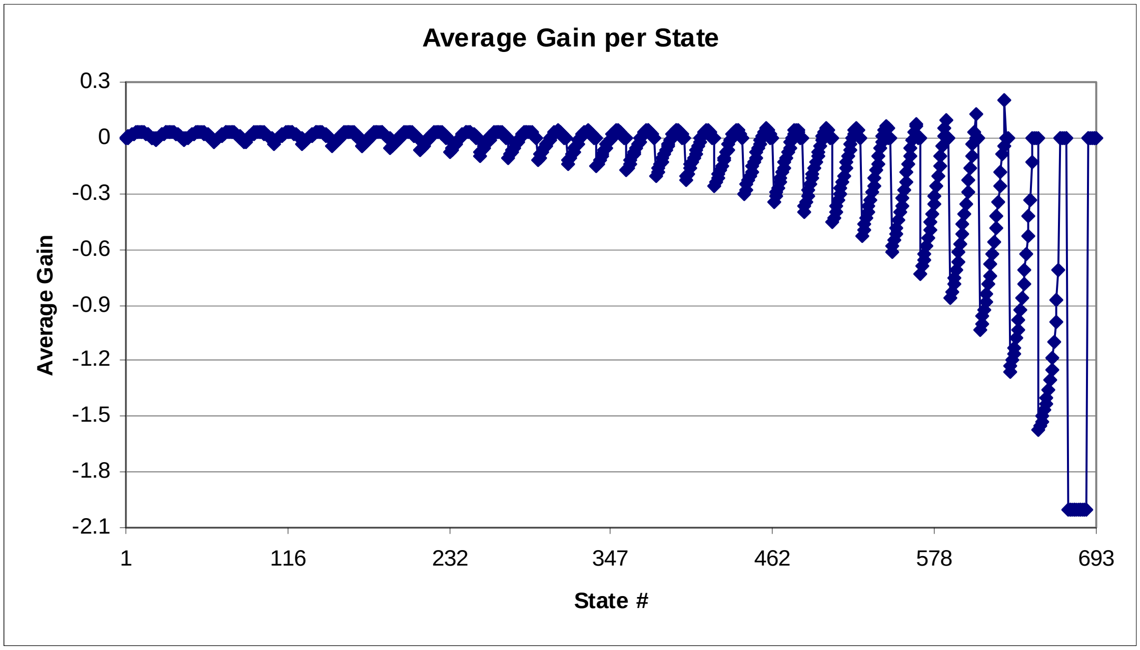

Figure 3.9: Effects of Removal on Full Deck - Average Gain vs. State #

As the deck is depleted of cards, the swings between positive gain and negative gain become more and more dramatic. Unfortunately for the player, the swings are not symmetric: the deep negative swings are not countered with a strong positive swing. The ability to know when the deck is in any of the positive areas of Figure 3.9 is desired. If this is possible, then the card player can simply alter his betting strategy without having to make any changes to the basic strategy. Figure 3.10 implies the card counter already knows where the positive areas in Figure 3.9 are:

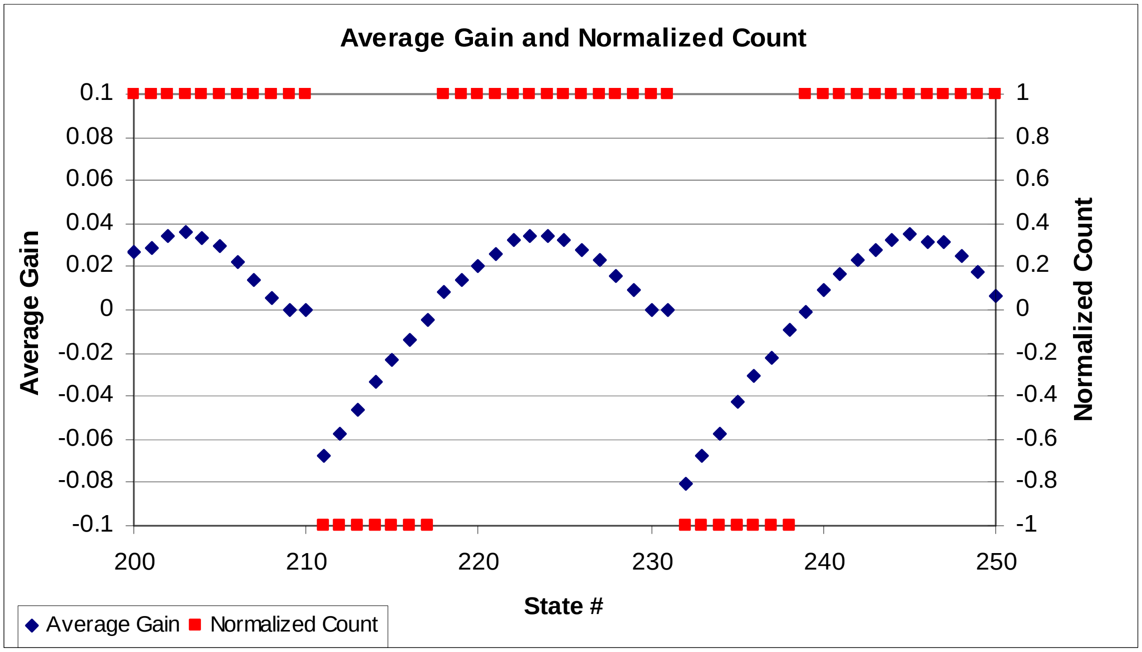

Figure 3.10: Average Gain plotted with the Normalized Count

Figure 3.10 illustrates the power of the card counting method. By zooming in on states 200 thru 250 from Figure 3.9 and plotting the Normalized Count on the secondary *y*-axis, it can be seen that an extraordinary amount of information about the deck is contained in the count.

Although states 200 thru 250 are a small subset of the total number of states, the count does, in fact, follow the same pattern for all the states: the count is positive in areas where the average gain is positive and the count is negative in areas where the average gain is negative. In Figure 3.10, state 239 has a slightly negative gain, but a positive count. One may question the validity of the count based on this. There are two things to keep in mind. First, the average gain is based upon simulation and still very small nonetheless. Second, the actual count on state 239 is +0.2.

There are a number of problems with defining the HMM with 693 states. One problem is that some states will simply never occur during play, such as state 2 (19, 32). The reason is that during play there will be between four and seven cards played on any given hand. Therefore, transitions will not occur between children as illustrated in Figure 3.8, but there will be jumps across states, as shown below in Figure 3.11. A detailed analysis (Appendix A) shows that a total of forty unique hands, corresponding to 11 different states, are possible for a single hand played with basic strategy from a full deck. In other words, there are only 11 states that can be reached from (20, 32) and they are shown in Figure 3.11.

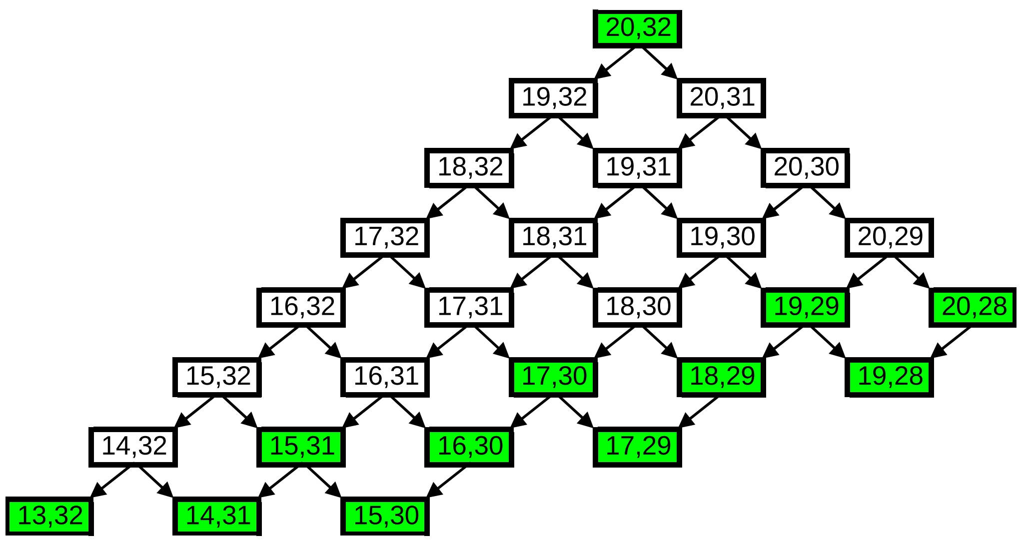

Figure 3.11: All possible states after first hand

The probability of transition from (20, 32) to (20, 28) is simply the probability of removing four tens from the deck times the number of ways to remove four tens. In other words

$$P\{(20,28)\,|\,(20,32)\} = \left(\frac{32}{52}\right)\left(\frac{31}{51}\right)\left(\frac{30}{50}\right)\left(\frac{29}{49}\right)\times 1 = 0.133 \tag{3.3}$$

Repeating the same calculation for the transition from (20, 32) to (19, 29), where there are four paths, leads to the following:

$$P\{(19,29)\,|\,(20,32)\} = \left(\frac{20}{52}\right)\left(\frac{32}{51}\right)\left(\frac{31}{50}\right)\left(\frac{30}{49}\right)\times 4 = 0.366 \tag{3.4}$$

However, there are actually only two valid paths from (20, 32) to (19, 29) based on basic strategy.

The transition from (20, 32) to (19, 29) occurs when one five and three tens have been removed from the deck. Only two possible hands satisfy this: player (F, T) vs. dealer (T, T) and player (T, F) vs. dealer (T, T) because basic strategy guides the player to stand on 15. On the other hand, were the player holding (T, T), that would mean the dealer was holding (T, F) or (F, T). Due to the structure of the game, the dealer does not stand on a total of fifteen: he must hit. The dealer will either draw a five ending the hand at (18, 29) or he will draw a ten ending the hand at (19, 28). Thus there are two valid paths from (20, 32) to (19, 29).

Therefore, in order to determine the transition vector for state 1, all possible states need to be found based on basic strategy play (Appendix A), and the transition probabilities need to be normalized according to how many valid paths exist. Such an analysis is not practical for over 600 states.

In an effort to reduce the number of states from 693 to something more practical, consider that each of the 693 states is actually a subset of a larger grouping that share similar characteristics. Based on data collected from simulations, the average gain for a single hand played with a deck of 16 fives and 32 tens is +1.638%. For 15 fives and 30 tens, the average gain is +1.669%. For 14 fives and 28 tens, the average gain is 1.728%. As the number of fives and tens decrease in such a manner, the average gain increases at each step. Therefore, it appears as though when the ratio of fives to the total number of cards in the deck is exactly 1/3, the deck is favorable to the player. R_{fT} is defined as the ratio of fives remaining in the deck to the total number of fives and tens in the deck, or

$$R_{fT} = \frac{f}{f+t} \tag{3.5}$$

To answer the question of whether R_{fT} behaves similarly for any f, t, a simple reordering of the data from Figure 3.9, sorted by R_{fT} is given in Figure 3.12.

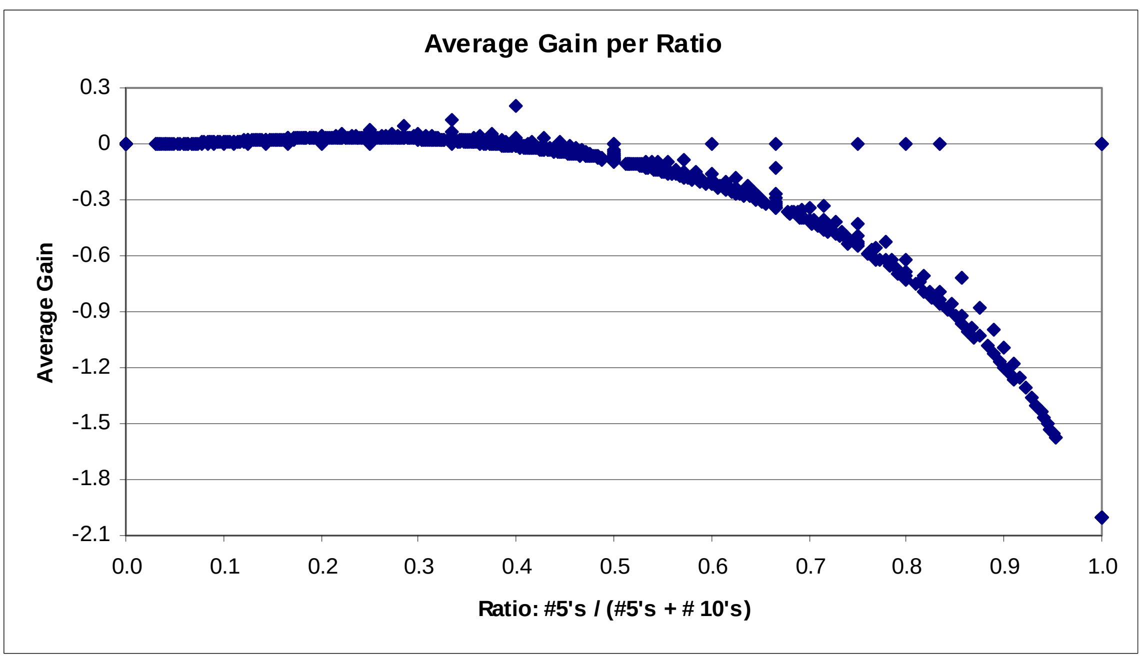

Figure 3.12: Average Gain vs. R_{fT}

Figure 3.12 does not represent the graph of a function. A number of ratios, such as ½, occur numerous times, but are not particularly visible on the scale. There also appear to be some outliers, such as an average gain of zero at $R_{fT} = 1$. This is due to exhaustion of the deck, where the gain is zero as defined earlier.

One data point of interest occurs on the *x*-axis at 0.4. An average gain of 19.94% from simulation occurs due to the distribution of the cards, which is 2 fives and 3 tens. For these five cards, there are a total of ten combinations possible. When analyzed in such a manner, of those ten combinations, two will result in a loss, four will result in a push, and four will result in a win. Therefore, the net player advantage is +2, or 20%.

Consider dividing Figure 3.12 into twelve distinct sections based on the value of R_{fT} as follows:

1) $R_{fT} = 0$
2) $0 < R_{fT} < 0.1$
3) $0.1 \leq R_{fT} < 0.2$
4) $0.2 \leq R_{fT} < 0.3$
5) $0.3 \leq R_{fT} < 0.4$
6) $0.4 \leq R_{fT} < 0.5$
7) $0.5 \leq R_{fT} < 0.6$
8) $0.6 \leq R_{fT} < 0.7$
9) $0.7 \leq R_{fT} < 0.8$
10) $0.8 \leq R_{fT} < 0.9$
11) $0.9 \leq R_{fT} < 1$
12) $R_{fT} = 1$

When R_{fT} is equal to zero it will remain as such until the deck has been exhausted. Thus, the player and the dealer will continue to push each hand with player's (T, T) vs. dealer's (T, T). However, at some point R_{fT} was not equal to zero, but eventually reached zero once all the fives were played out of the deck. In other words, $R_{fT} = 0$ could represent a state of the deck and a non-zero value, or range of values of R_{fT} could represent a different state.

On average, each of the twelve distinct sections defined above behaves in a similar matter. Thus those twelve sections could each represent a state in a finite-state automaton. As the deck is depleted of cards, R_{fT} changes. Sometimes, R_{fT} remains inside any one of the twelve boundaries described above and sometimes it crosses over a certain threshold and enters a new state. If the transition probabilities between the states can either be determined or measured, then this system could be described as a Markov chain. Table 3.2 defines the state number given the value of R_{fT}. Note that numbering, or even the naming of states, is arbitrary.

Table 3.2: Definition of States

Value of R_{fT}	**State #**
$R_{fT} = 0$	5
$0 < R_{fT} < 0.1$	4
$0.1 \leq R_{fT} < 0.2$	3
$0.2 \leq R_{fT} < 0.3$	2
$0.3 \leq R_{fT} < 0.4$	1
$0.4 \leq R_{fT} < 0.5$	6
$0.5 \leq R_{fT} < 0.6$	7
$0.6 \leq R_{fT} < 0.7$	8
$0.7 \leq R_{fT} < 0.8$	9
$0.8 \leq R_{fT} < 0.9$	10
$0.9 \leq R_{fT} < 1$	11
$R_{fT} = 1$	12

By defining R_{fT} to be the ratio of fives to total cards in the deck and plotting the average gain from Figure 3.9 against R_{fT}, a set of states were defined in Table 3.2. Knowing the transition probabilities between the states in Table 3.2 would lead to a Markov chain. Thus, instead of defining the state of the deck as one of 693 states based on the distribution of cards, the state of the deck can be defined by R_{fT}.

The next step in defining an HMM is determining what the observations are. The card-counter essentially keeps a running total of the number of fives and tens played from the deck, so it may make sense to do something similar for the observation after a hand has been completed. Therefore, define the number of fives exposed after a hand has been completed as the observation. Thus, for state 5, where $R_{fT} = 0$, the observation will always be zero fives and the transition will always be back to state 5.

Finally, completion of the HMM definition requires a prior probability vector. As was previously defined when there were 693 states, the starting state is the full deck of cards and the prior vector assigns a value of one to the entry corresponding to the initial state. As defined in Table 3.2, state 1 is for $0.3 \leq R_{fT} < 0.4$. For 20 fives and 32 tens, R_{fT} is equal to 0.385, hence the assignment in Table 3.2.

3.4 HMM Training

With the states defined based on R_{fT} and the observations defined as the number of fives seen after a hand, the HMM parameters need to be determined. That is, the transition and observation matrices need to be found. As was discussed in section 3.3, for state 5 ($R_{fT} = 0$) the observation will always be zero fives and the transition will always be back to state 5. How other states behave is clearly a different matter and a look at Figure 3.13 will show why.

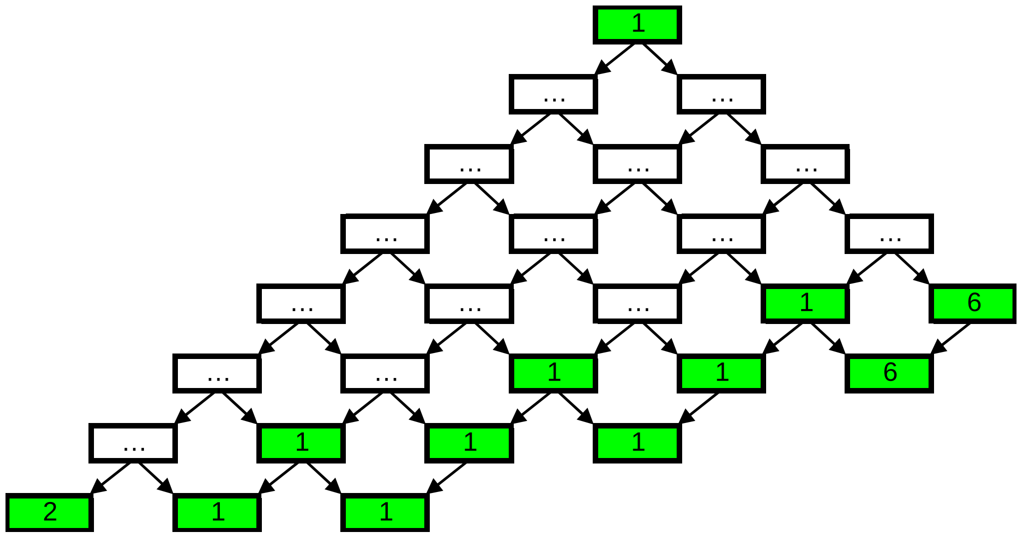

Figure 3.13: Possible states after first hand

Beginning with a full deck (20 fives, 32 tens), Figure 3.13 implies that the only possible transitions are to states 1, 2, and 6. Thus, a transition vector for state 1 would have non-zero values for states 1, 2, and 6 and zero for the remaining states.

Consider the deck in a configuration of 4 fives and 7 tens where $R_{fT} = 0.364$, corresponding to state 1. One possible hand that could be played would result in player (T, T) vs. dealer (T, T). This would leave the deck in a configuration of 4 fives and 3 tens for value of $R_{fT} = 0.571$, corresponding to state 7. This clearly contradicts the previous paragraph, which had a zero probability of transition from state 1 to state 7.

Unfortunately, since the deck is being depleted of cards, the behavior of state 1 when there are 52 cards is different than the behavior of state 1 when there are 11 cards. Therefore, to determine the transition vector for state 1, it would be necessary to take all card combinations of fives and tens (f, t) where $0.3 \leq R_{fT} < 0.4$ and determine the subsequent orderings of the deck after a single hand of play as was done for (20, 32). There are 122 orderings of (f, t) for $0.3 \leq R_{fT} < 0.4$. Once the various transition probabilities were found and properly weighed, a transition vector would be complete for state 1. The process would then continue for the remainder of the states until a transition matrix is complete.

A less intensive way to determine the transition vector for, say, state 1 would be to choose a subset of the 122 orderings and allow those to be representative of them all.

An even better way to determine the transition matrix would be to simply train the HMM parameters. Unfortunately, the traditional HMM training algorithms do not work with this due to the structure of this problem. However, development and implementation of an ad hoc training algorithm will ultimately give fully defined transition and observation matrices.

At this point, it is necessary to update Figure 3.12, which has additional information that needs to be extracted. Recall from Figure 3.11 that not all combinations of fives and tens will be encountered. Also, recall from Table 3.1 there are 19 exhausted hands. Removal of these additional deck configurations results in Figure 3.14, which is zoomed in on the y-axis from -0.1 to 0.1.

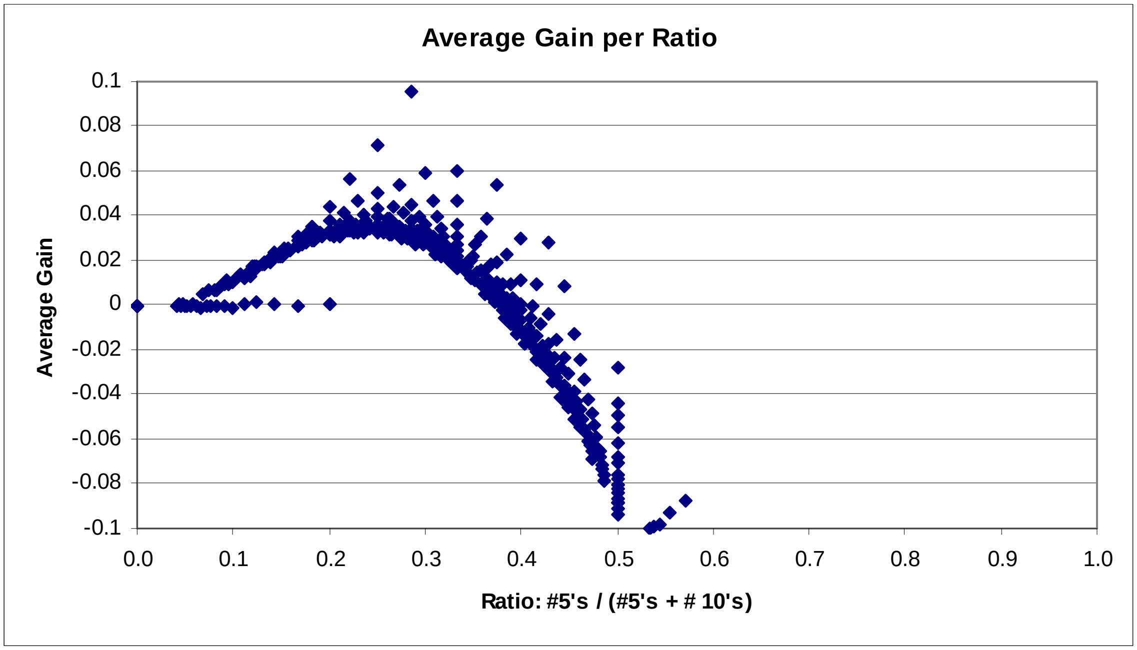

Figure 3.14: Update of Figure 3.12.

From Figure 3.14, it is clear that there are values of R_{fT} where the average gain is clearly positive and clearly negative. For $0.135 \le R_{fT} < 0.33$, the average gain is at least 2%. This is an area the player clearly wants to take advantage of. Rather than using twelve states as defined in Table 3.2, consider using eight states as defined in Table 3.3.

Table 3.3: Definition of Hidden States

Value of R_{fT}	State #
$0 \le R_{fT} < 0.135$	6
$0.135 \le R_{fT} < 0.33$	5
$0.33 \le R_{fT} < 0.35$	4
$0.35 \le R_{fT} < 0.365$	3
$0.365 \le R_{fT} < 0.38$	2
$0.38 \le R_{fT} < 0.4$	1
$0.4 \le R_{fT} < 0.5$	7
$0.5 \le R_{fT} \le 1$	8

Once again, state 1 was chosen such that the value of R_{fT} corresponded to the full deck of cards. State 5, where $0.135 \leq R_{fT} < 0.33$, is the ideal state the card player desires to be in.

Once again, the observations are the number of fives after each hand is played. This value will vary from zero to seven.

Up to this point, abbreviated tree diagrams have been shown for the distributions of fives and tens. Now, with the definition of the hidden states complete, a visual of the entire tree diagram is shown in Figure 3.15. Note that there are no arrows pointing to the children. Instead, the children of each distribution (f, t) are represented by the touching of the corners of each box. The states are color-coded, with the definitions given in the Figure.

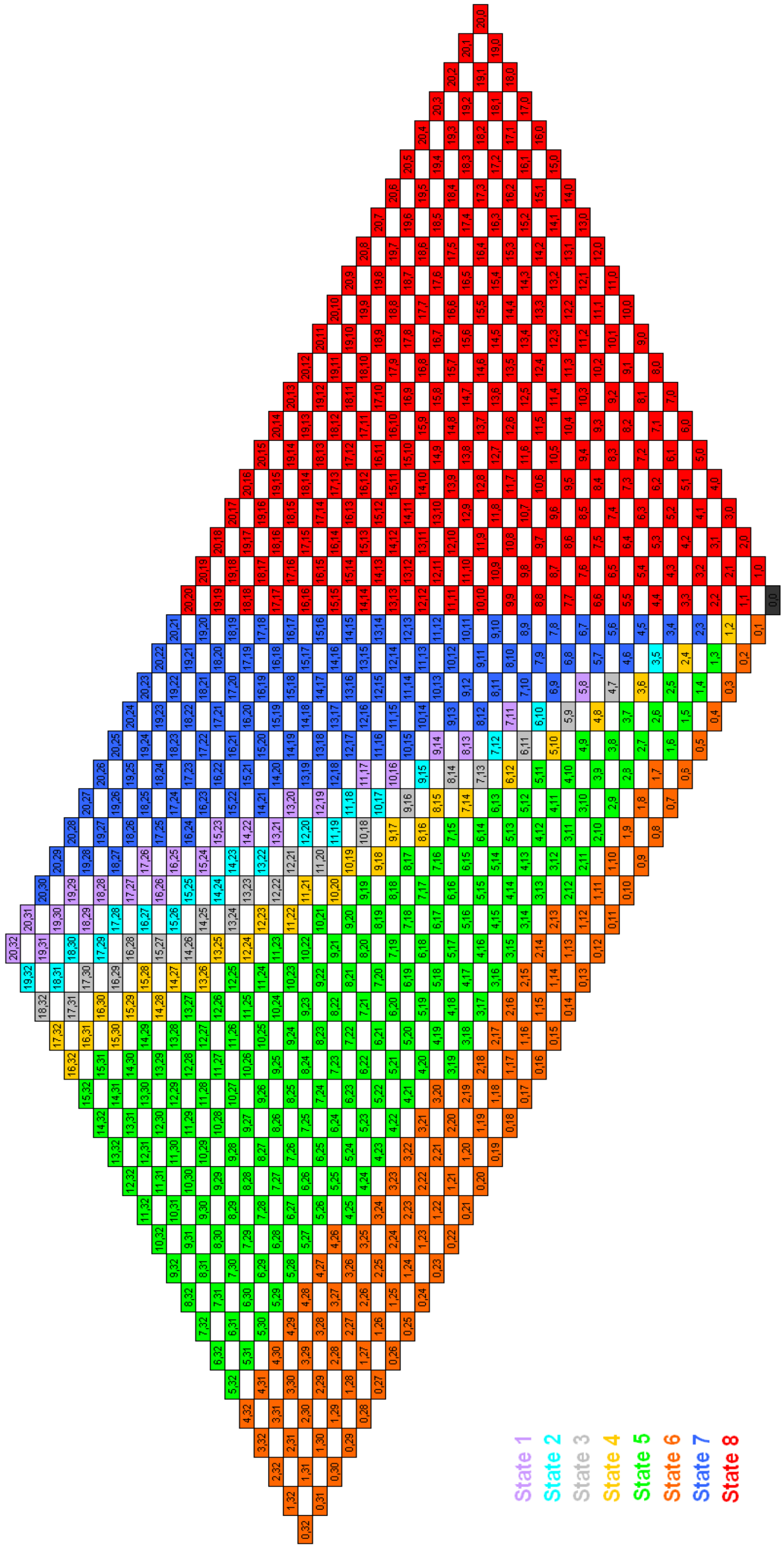

Figure 3.15: Tree diagram of all 693 distributions of the deck, color-coded for the 8 hidden states.

There are 693 possible distributions of a deck containing 20 fives and 32 tens. Simulation of each configuration of the deck gives the average gain for any number of fives and tens. By taking the ratio of fives to total cards in the deck, the areas of play that are advantageous were found. Dividing these ratios into distinct areas gives Figure 3.15, which takes the 693 distributions and turns them into 8 states of a Hidden Markov Model.

An ad hoc training method is performed via simulation of basic strategy play. By tracking the states, the number of transitions, and the observations, the transition and observation matrices can be determined. Earlier it was discussed that 9 hands could be played 100% of the time while tens hands could be played about 99.96% of the time. For the purposes of simplification of the ad hoc training algorithm, the number of hands played will be limited to 9.

The ad hoc training algorithm is implemented as follows. When complete, the observation and transition matrices will be known.

1. For each of the 8 states:
    a. Initialize a 1x8 transition vector, S_i = [0 0 0 0 0 0 0 0]
    b. Initialize a 1x8 observation vector, O_i = [0 0 0 0 0 0 0 0]
2. Shuffle the Woolworth Deck
3. Calculate R_{fT} and determine the current state, k
4. Play a hand according to basic strategy and record the observation, v_k
5. Calculate R_{fT} and determine the next state, $k + 1$
6. For S_k, increment the column corresponding to $k + 1$ by one
7. For O_k, increment the column corresponding to v_k by one
8. The next state becomes the current state. Return to step 4.

9. Play a total of nine hands. After the 9th hand, return to step 2.

10. Repeat the entire procedure for *D* decks.

11. Each entry in S_i will be the number of times state *j* transitioned to it.

    a. Determine the relative frequency for S_i by dividing each entry by the sum of all entries.

12. Each entry in O_i will be the number of times state *i* emitted *v*.

    a. Determine the relative frequency for O_i by dividing each entry by the sum of all entries.

13. Combine [S_1, S_2, …, S_8] to form a transition matrix. Combine [O_1, O_2, …, O_8] to form an observation matrix.

Upon completion of the ad hoc training, the resultant transition and observation matrices are given below:

$$A = \begin{bmatrix} .3847 & .0778 & .1268 & .0659 & .0367 & 0 & .3044 & .0037 \\ .1681 & .1878 & .1359 & .1236 & .1186 & .0002 & .2633 & .0025 \\ .1754 & .1744 & .2078 & .1628 & .1828 & .0005 & .0963 & 0 \\ .1008 & .1137 & .1783 & .2102 & .3358 & .0007 & .0606 & 0 \\ .0235 & .0240 & .0714 & .1549 & .6981 & .0186 & .0094 & 0 \\ 0 & 0 & 0 & 0 & .4293 & .5707 & 0 & 0 \\ .0989 & .0534 & .0381 & .0250 & .0193 & 0 & .6516 & .1136 \\ .0153 & .0019 & .0095 & .0054 & .0028 & 0 & .2586 & .7065 \end{bmatrix} \tag{3.6}$$

$$B = \begin{bmatrix} .1266 & .2916 & .2980 & .1969 & .0626 & .0215 & .0023 & .0006 \\ .1387 & .3068 & .2988 & .1841 & .0541 & .0157 & .0017 & .0002 \\ .1480 & .3241 & .2995 & .1697 & .0454 & .0121 & .0011 & .0002 \\ .1655 & .3456 & .2944 & .1505 & .0352 & .0081 & .0007 & .0001 \\ .2397 & .3928 & .2565 & .0925 & .0159 & .0026 & .0001 & 0 \\ .6466 & .3323 & .0211 & 0 & 0 & 0 & 0 & 0 \\ .0865 & .2499 & .3000 & .2404 & .0859 & .0328 & .0037 & .0009 \\ .0320 & .1488 & .2530 & .3138 & .1518 & .0848 & .0126 & .0033 \end{bmatrix} \quad (3.7)$$

Of course, the HMM always begins in state 1 with a freshly shuffled deck. Therefore, the prior vector takes on the following form:

$$\pi = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.8)$$

With the HMM fully defined, it can now be applied to the game of Woolworth Blackjack.

3.5 HMM Strategy

The HMM Strategy is the use of a Hidden Markov Model and associated analysis tools for the purpose of gaining an advantage in the game of Woolworth Blackjack.

During play, as the deck is depleted of cards during play, R_{fT} changes. Ideally, when the HMM is in state 5, i.e. R_{fT} is between 0.135 and 0.33, the card player wants to take advantage of this and bet heavily in this region. Otherwise, the player should shrink the size of his bets or refrain from betting altogether when in any of the other states.

Consider what happens after the first hand has been played, where the deck begins in state k. First, the player will make an observation: the number of fives dealt during the hand. Second, R_{fT} will have changed and the deck will be in state $(k + 1)$. Two pieces of information are needed at this point: the previous state, *k*, and the current state (*k* + 1). The Viterbi Algorithm will be employed to determine *k*.

Figure 3.16 illustrates this with an example where the first hand played with basic strategy results in player (F, T) vs. dealer (T, F, T):

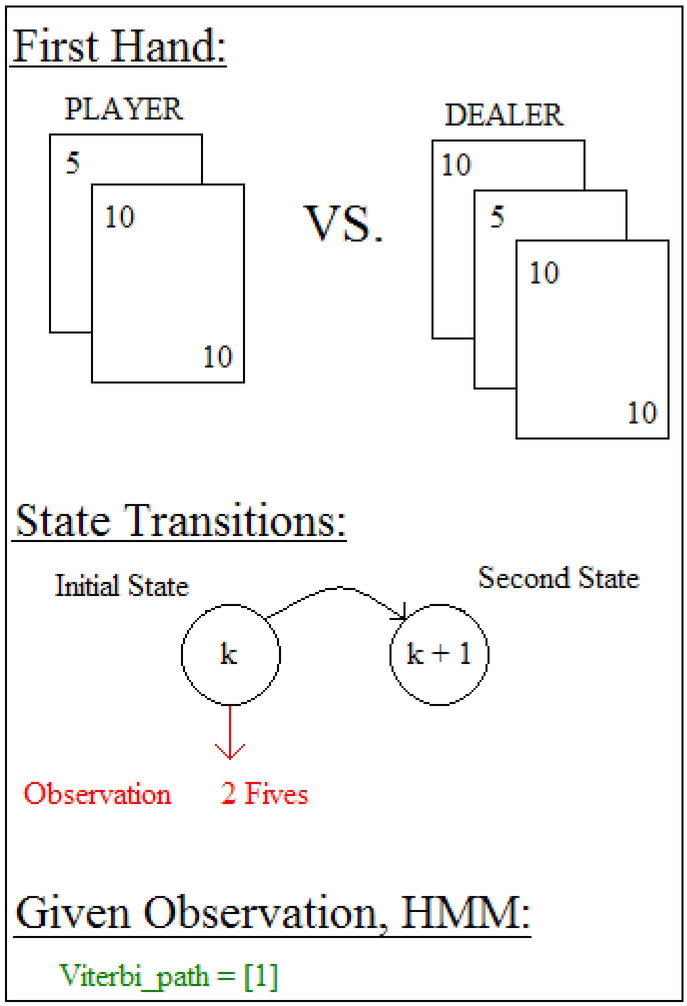

Figure 3.16: Single hand evaluated with HMM Tools

Beginning in state *k*, an observation of two fives is made. Therefore, the observation vector is $V = [2]$. From the prior probability vector, it is already known that the HMM begins in state 1. Nevertheless, when the transition matrix, the observation matrix, the prior probability vector, and the observation vector are fed into the Viterbi Algorithm, the result will be *Viterbi_path* = [1], meaning k = state 1.

Consider Figure 3.17, where a second hand is played with basic strategy resulting in player (T, T) vs. dealer (T, T):

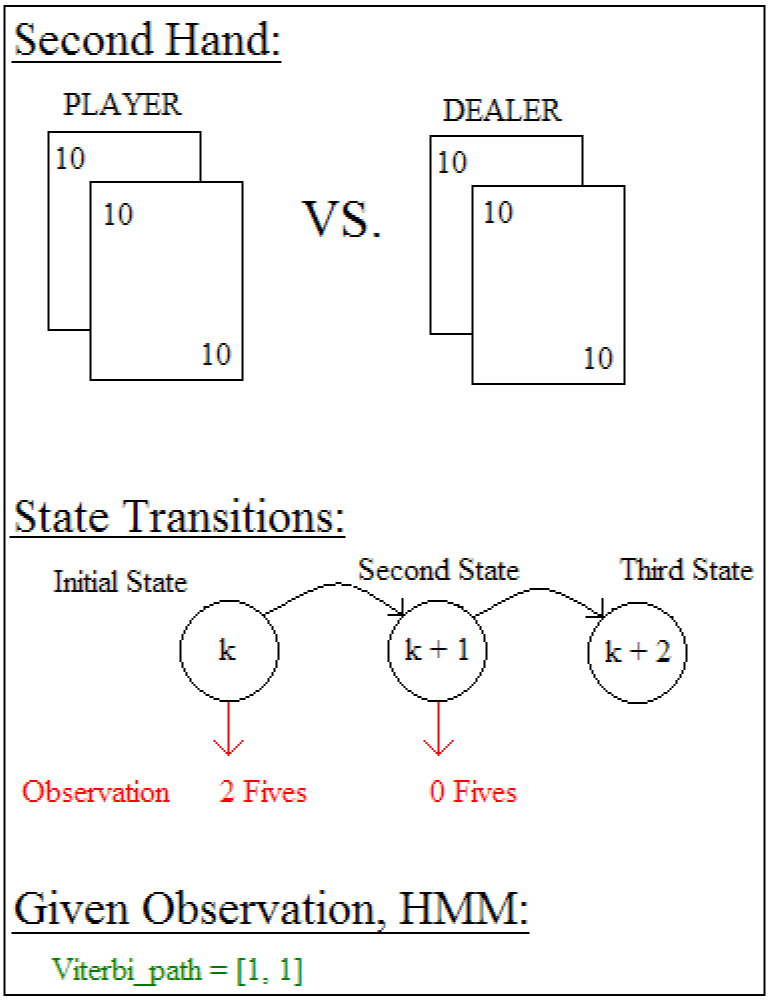

Figure 3.17: 2 hand example

The previous observations are not ignored. Therefore, beginning in state k, an observation of 2 fives is made. In state k + 1, an observation of zero fives in made. Therefore, the observation vector is V = [2 0]. Once again, the transition matrix, the observation matrix, the prior probability vector, and the observation vector are fed into the Viterbi Algorithm. The result in this case is a *Viterbi_path* = [1 1], meaning k = state 1 and $k + 1$ = state 1.

As was previously stated, there are two pieces of information are needed: the previous state, k, and the current state $k + 1$. The Viterbi Algorithm is used to determine k. More generally, when n hands have been played, the Viterbi Algorithm will return a path corresponding to the most likely states that generated the observations. That is, the Viterbi Algorithm will return

$$Viterbi\_path = [(k-n) \;\; \dots \;\; (k-2) \;\; (k-1) \;\; k] \tag{3.9}$$

What is desired is the current state, $(k + 1)$. Knowledge of the current state is what the player uses to determine how to wager.

One method to determine state $(k + 1)$ would be to simply look at row k of the transition matrix. The maximum value in row k represents the most likely transition from state k. This would be state $(k + 1)$.

To determine the validity of this, consider the following example. Suppose there are four urns, each containing a unique deck of cards that has a fixed R_{fT}. Urn # 1, has 7 fives and 13 tens ($R_{fT} = 0.35$). Urn # 2 has 3 fives and 17 tens ($R_{fT} = 0.15$). Urn # 3 has 9 fives and 11 tens ($R_{fT} = 0.45$). Finally, urn # 4 has 15 fives and 5 tens ($R_{fT} = 0.75$). Based on simulation, the average gain for a single hand of Woolworth Blackjack played from each of the decks in urn 1, urn 2, urn 3, and urn 4 are +2.47%, +2.34%, -3.19%, and -54.2%, respectively.

After training of each deck, the following observation matrix is determined:

$$
B_U = \begin{bmatrix} .146 & .336 & .310 & .164 & .037 & .007 & 0 & 0 \\ .493 & .393 & .105 & .009 & 0 & 0 & 0 & 0 \\ .068 & .231 & .308 & .260 & .095 & .034 & .004 & 0 \\ .026 & .130 & .239 & .321 & .164 & .100 & .015 & .005 \end{bmatrix} \tag{3.10}
$$

The following arbitrarily defined transition matrix is given:

$$
A_U = \begin{bmatrix} .2 & .1 & .7 & 0 \\ .35 & .25 & .1 & .3 \\ .1 & .4 & .3 & .2 \\ .2 & 0 & .3 & .5 \end{bmatrix} \tag{3.11}
$$

Finally, the prior probability vector is π_U = [1 0 0 0].

Imagine an elaborate game of Woolworth Blackjack where the dealer deals a single hand from one of the four urns. The game begins when the dealer uses π_U to determine the first deck to be used. The deck is returned to the proper urn after one hand has been played and the settlement is complete. The dealer then selects the next deck to deal from based on A_U. Play continues until a set number of hands are played and the entire process repeats.

In such a scenario, it is clear that the counting strategy does not apply. However, the HMM strategy is employed. After the first hand has been played, the strategy requires the knowledge of state k as well as the current state, $k + 1$. The Viterbi Algorithm is used to determine k. Once k is known, then the maximum value in row k of the transition matrix will correspond to state ($k + 1$). Since the average gain for each deck is known, the player using the HMM strategy can act accordingly: bet when decks (urns) 1 and 2 are being used and refrain from betting when decks 3 and 4 are being used.

Simulation results of the game comparing the HMM strategy to the player simply using the basic strategy are given in Figure 3.18. A total of 9 hands are played. After the ninth hand, the game starts over. That is, after the ninth hand the dealer uses the prior probability vector to select the deck for play. This mimics shuffling of a Woolworth deck and ensures observation vectors and Viterbi paths each have nine or less items.

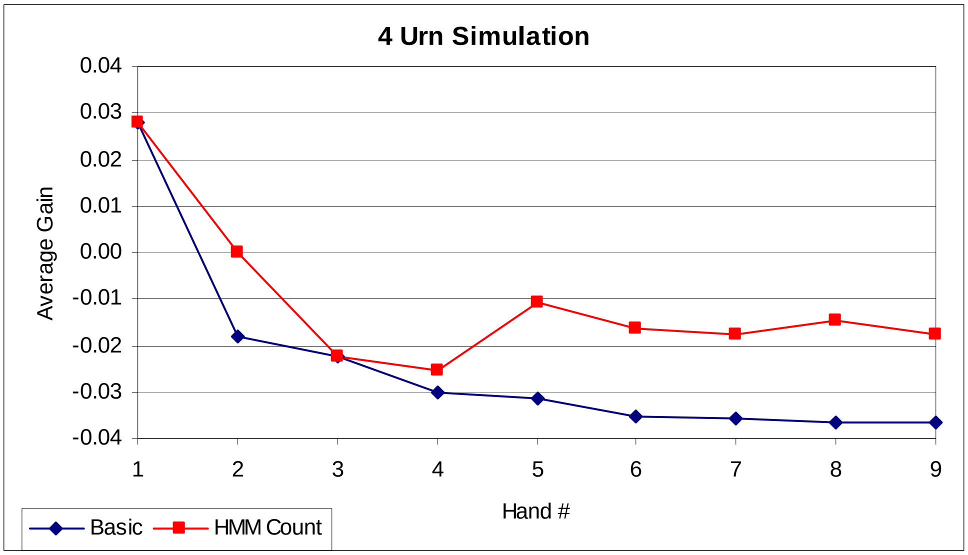

Figure 3.18: HMM strategy applied to the Urn example

Clearly the HMM strategy is an improvement over the basic strategy. So long as a single Woolworth deck behaves in a manner similar to this urn example, the HMM strategy will certainly be an improvement over the basic strategy. However, the performance relative to the counting strategy is yet to be seen.

Unfortunately, for play with the Woolworth deck, determination of state (k + 1) based on the maximum value of row k in the transition matrix is not valid. To see why this is, consider the transition vector for state 1 and Figure 3.19, which represents the possible states after the first hand has been dealt:

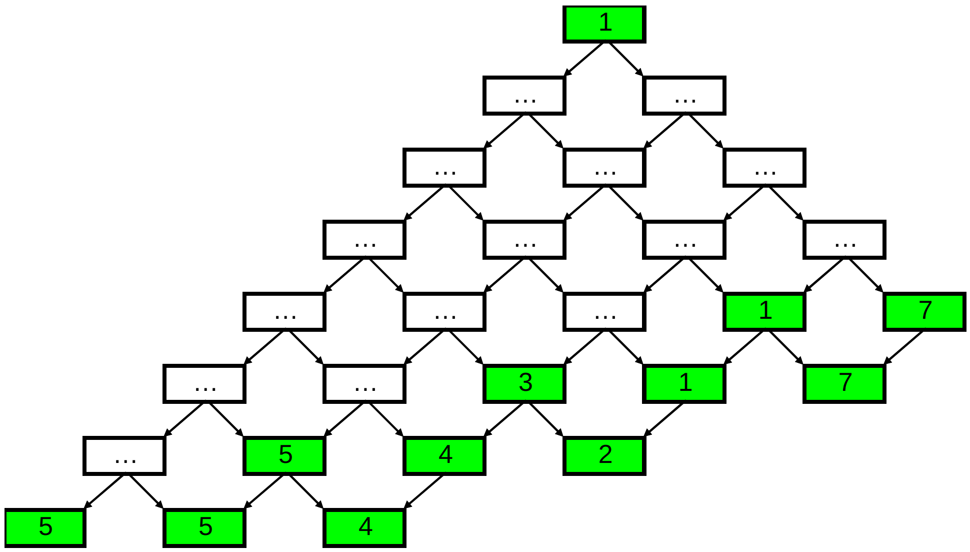

Figure 3.19: Possible states after first hand

$$A_1 = \begin{bmatrix} .3847 & .0778 & .1268 & .0659 & .0367 & 0 & .3044 & .0037 \end{bmatrix} \tag{3.12}$$

Note that Figure 3.19 is an update of Figure 3.13 based on the states defined in Table 3.3. Suppose the following hand is dealt: player (T, T) vs. dealer (T, T). In this case, the observation is zero fives. From the HMM parameters, *Viterbi_path* = [1] (see Figure 3.16). Therefore the current state, state (k + 1) is determined from the maximum value in A_1. The maximum value is 0.3847, which corresponds to state 1. However, based on Figure 3.19 this clearly cannot be. R_{fT} = (20)/(20 + 28) = 0.417, which corresponds to state 7.

The reason this problem does not occur in the 4 urn problem is because the transition is completely independent of the observation, as is the expected behavior of a HMM. Unfortunately, when a single deck of cards is modeled as a HMM, there is a dependency on the observation because the actual structure of the deck is changing when the cards are removed.

To address this dependency, the transition dependency, T_d, is defined as the probability that state k will transition to state $(k + 1)$ given observation O.

$$T_d = P\{(k+1) \mid k, O\} \tag{3.13}$$

The transition dependency is defined as a matrix for each state of the HMM. The parameters for the matrices are trained using ad hoc methods. Figure 3.20 outlines the structure of the transition dependency matrices.

NEXT STATE:

$$\text{OBSERVATIONS:}\quad \begin{array}{r} \\ \text{0 Fives} \\ \text{1 Five} \\ \text{2 Fives} \\ \text{3 Fives} \\ \text{4 Fives} \\ \text{5 Fives} \\ \text{6 Fives} \\ \text{7 Fives} \end{array} \begin{array}{c} \begin{array}{cccccccc} \textit{State } 1 & \textit{State } 2 & \textit{State } 3 & \textit{State } 4 & \textit{State } 5 & \textit{State } 6 & \textit{State } 7 & \textit{State } 8 \end{array} \\ \begin{bmatrix} t_{01} & t_{02} & & & \cdots & & & t_{08} \\ t_{11} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ \vdots & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ t_{71} & & & & \cdots & & & t_{88} \end{bmatrix} \end{array}$$

Figure 3.20: Transition Dependency Matrix structure

The transition dependency matrix is used to determine the most probably current state given the previous observation. In other words, it is used to determine state $(k + 1)$ given state k and the observation generated by state k. The transition dependency matrix for state 1 is given below:

$$T_d[1] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & .971 & .029 \\ .379 & 0 & 0 & 0 & 0 & 0 & .621 & 0 \\ .914 & .086 & 0 & 0 & 0 & 0 & 0 & 0 \\ .017 & .266 & .595 & .090 & .032 & 0 & 0 & 0 \\ 0 & 0 & .151 & .175 & .134 & 0 & 0 & 0 \\ 0 & 0 & 0 & .150 & .850 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .989 & .011 & 0 & 0 \\ 0 & 0 & 0 & 0 & .989 & .011 & 0 & 0 \end{bmatrix} \tag{3.14}$$

From the example given above, when beginning in state $k = 1$, if an observation of zero fives is made, then the maximum value of $T_d[1]$ is found to be 0.971. This corresponds to state 7, which agrees with Figure 3.19. Note the probability of transition to State 8 is 0.029. This is due to certain distributions of the deck such as 5 fives and 8 tens. In such a case, $R_{fT} = (5)/(5 + 8) = 0.385$, which corresponds to state 1. A hand played (T, T) vs. (T, T) leaves 5 fives and 4 tens for $R_{fT} = (5)/(5 + 4) = 0.556$, which corresponds to state 8.

With the transition dependency matrices defined for each state, the HMM strategy can now be applied to the deck. The HMM strategy is implemented as follows:

1. Play begins with a standard, well-shuffled 52 card Woolworth deck
2. A single hand is played using basic strategy
3. After the hand is complete, the observation is the number of fives exposed
4. HMM parameters, including the observation, are fed into the Viterbi Algorithm to determine the previous state
5. Given the previous state and observation, the transition dependency matrix is used to determine most probably current state
6. If the average gain for the current state is positive, a wager is placed. Otherwise, no wager is placed.
7. The process repeats for a total of nine hands

In order to simplify the betting, the HMM strategy will either place a wager of one unit or refrain from betting. The hand is played whether or not a wager is placed.

To perform a comparison of the basic strategy, the counting strategy, and the HMM strategy, the simulation will be setup as follows. A single "community" hand will be played against the dealer's hand. It will be played strictly using the basic strategy. A total of nine hands will be played without replacement of cards or reshuffling. Each player will bet one unit on the first hand, and the basic strategy player will wager on all nine hands. For the second thru ninth hands, the counting strategy player will place a wager when the count is positive and will not wager otherwise. The HMM strategy player will place a wager when the deck is in state five, i.e. $0.135 \leq R_{fT} < 0.33$ and will not wager otherwise. The average gain for each of the three players is shown in Figure 3.21.

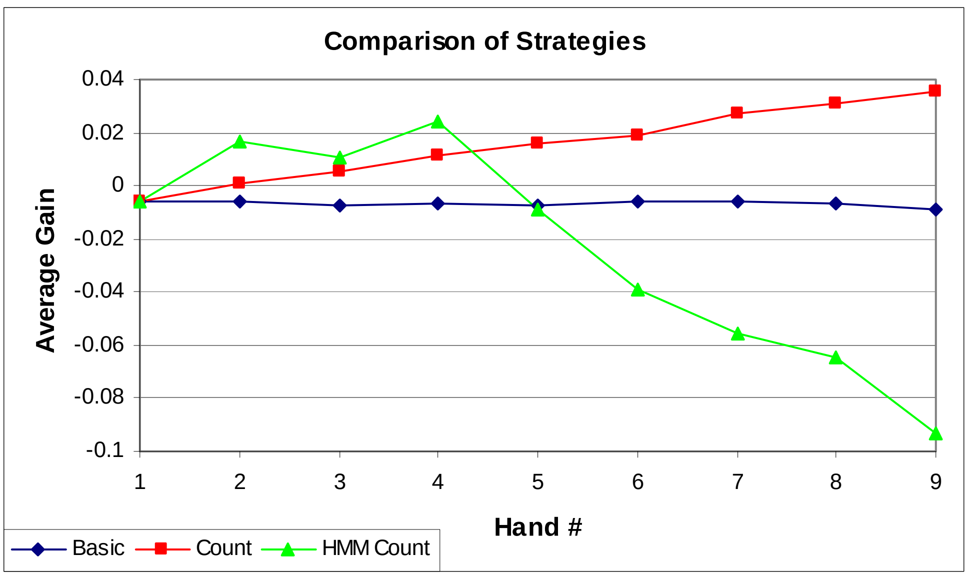

Figure 3.21: Comparison of Basic, Counting, and HMM strategies

For the first hand played, it is obvious that the average gain for the three players is the same because they all place a wager on the first hand. The counting strategy shows a consistent advantage over the basic strategy, as was shown earlier in Figure 3.5. For hands 2, 3, and 4, the HMM strategy is superior to both. On the other hand, the HMM strategy is inferior to both strategies from the 5^{th} hand on.

The HMM strategy is superior in the early hands because state 5 was chosen based on R_{fT} that had a 2% or better advantage. On the other hand, as was shown in Figure 3.10, the counting strategy is often employed in areas where the advantage is very small. Thus the lower rate of return.

As the deck is depleted of cards, a number of things happen. First, as the number of observations increase, the amount of uncertainty in the Viterbi path increases. As the uncertainty in the Viterbi path increases, the uncertainty in the transition dependency matrix increases as well. Since there are more favorable states than unfavorable states, the player will tend to place more and more wagers in unfavorable areas when he should refrain and he will refrain assuming he is in an unfavorable situation when he is in fact not.

Of course, the states simply behave differently when the number of cards in the deck changes. Because of this, a training method was devised to simply group the behavior or R_{fT} for any number of cards. But clearly the behavior when there are 20 fives and 32 tens differs than when there are 5 fives and 8 tens. This leads to even greater uncertainty about the true state of the deck as it is depleted of cards.

Chapter 4. Conclusions and Future Work

The card counting method developed for Woolworth Blackjack is clearly a powerful tool. The card counter has an incredible amount of information before him when he counts. A side-by-side comparison of the count vs. the average gain (Figure 3.10) shows that both are either negative or positive.

Modeling a deck of cards as a HMM was not a trivial task. A number of liberties and constraints had to be taken in order to do so. For the game of Woolworth Blackjack, the deck of cards was modeled based on R_{fT}. By knowing the areas of the deck that are favorable, based on simulation, HMM tools could be used to determine when the deck was in such an area and the player could take advantage of this.

One approach to improving the HMM count would be to re-visit the training method. Due to the depletion of cards from the deck, the training weighs heavily on the first part of the deck. As the deck is depleted of more and more cards, the HMM parameters may not actually match what the bottom of the deck is doing.

Clearly with the deck beginning in state 1, the transition probabilities for state 1 at the beginning of the deck are quite strong. However, towards the end of the deck, the transition probabilities for state 1 may no longer apply. Therefore, instead of playing 9 hands from a full deck for training, perhaps starting at the middle of the deck and playing 2 – 5 hands would help improve the performance at the end.

Another area of improvement could be choice of states. Initially, five states were chosen, as follows:

1) $0 \leq R_{fT} < 0.135$

2) $0.135 \leq R_{fT} < 0.33$

3) $0.33 \leq R_{fT} < 0.4$

4) $0.4 \leq R_{fT} < 0.5$

5) $0.5 \leq R_{fT}$

The idea behind the five states was as follows: 0.135 to 0.33 is the optimal area, where the player desired to be and the average gain was at least 2%. Ignore the areas outside of it. However, at $R_{fT} < 0.135$ there is an excess of tens in the deck and more than likely, the tens will come out at a faster pace than the fives. Once R_{fT} reaches 0.5, the average gain is clearly terrible, but placing state 1 between 0.33 and 0.5 seemed like quite a large gap. Thus, two “buffer” zones were entered between 0.33 and 0.5.

However, after training the HMM based on five states, there was a huge problem when the HMM strategy was performed. The Viterbi algorithm would continually return a path indicating *all* previous states were state 1. Furthermore, the transition dependency matrix for state 1 would indicate the most probable next state to be state 1: a state that was to be avoided. Thus, the HMM strategy literally never found situations where the deck was favorable.

Therefore, fives states was changed to eight. The change happened by increasing the area from $0.33 \leq R_{fT} \leq 0.5$ from two states to five. This allowed the Viterbi algorithm to return something other than a sequence of ones and the transition dependency matrices would actually indicate favorable states.

Redefining the states or changing the number of states coupled with improved training methods are two ways this counting method could be improved.

Applying this methodology to traditional Blackjack would be quite an undertaking. There are a number of issues that arise. First, a Blackjack deck is far more complex than a Woolworth deck. The change is from a 52 card deck with cards of two differing ranks (fives and tens) to a 52 card deck with cards of eleven different ranks (face cards are ranked 10 and the Ace can be 1 or 11).

Next, the additional cards make Blackjack a far more complex game. Blackjack involves pair-splitting, doubling down, doubling down after splitting, surrender, and insurance to name a few.

However, it is conceivable that the game could be narrowed down to some specific rules, such as forbidding pair splitting and only doubling-down on totals of ten or eleven. From there, a basic strategy could be devised. Next, simulations could be run of the entire deck to determine the average gain. Then, certain cards (such as all the twos) could be removed from the deck to determine the effect on the gain.

It may turn out that some cards have a positive effect on the gain, some a negative effect on the gain, and some no effect. Thus an HMM could be defined based on the ratio of a subset of the thirteen different cards, while the others remain neutral. From there, it is a matter of determining the observations, training the HMM, and applying it to the game.

REFERENCES

1. Moon, Todd and Wynn Stirling. *Mathematical Methods and Algorithms for Signal Processing*. Prentice Hall, 2000.
2. Stark, Henry and John Woods. *Probability and Random Processes with Applications to Signal Processing*. Prentice Hall, 2001.
3. Fink, Gernot A. *Markov Models for Pattern Recognition*. Springer, 2010.
4. Duda, Hart & Stork. *Pattern Classification*. Wiley-Interscience, 2000.
5. Rabiner, L.R. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, Vol. 77, No. 2, February 1989.
6. Rabiner, L.R. and B.H. Juang. An Introduction to Hidden Markov Models. *IEEE ASSP Magazine*, January 1986.
7. Forney, G. David Jr. The Viterbi Algorithm. *Proceedings of the IEEE,* Vol. 61, No. 3, March 1973.
8. Thorp, Edward O. *Beat the Dealer*. Vintage, 1966.
9. Griffin, Peter A. *The Theory of Blackjack*. 6th edition, Huntington Press 1999.
10. Thorp, Edward O. and William E. Walden. The Fundamental Theorem of Card Counting with Applications to Trente-et-Quarante and Baccarat. *International Journal of Game Theory,* Vol. 2, No. 1, pp. 109-119.
11. Morehead, Albert H. et al. *Hoyle's Rules of Games*. 3rd edition, Signet 2001

Appendix A. First Hand Analysis

For a standard 52-card Woolworth Blackjack deck, there are f = 20 fives and t = 32 tens. This is represented as (20, 32). After a single hand has been played, the deck will be in a configuration of ($f - k$, $t - n$), where k and n represent the number of fives and tens removed from the deck, respectively. There will be a total of N configurations of the deck after a single hand of play.

It is desired to know what the N configurations of the deck will be after the first hand. In order to determine the N configurations, all possible hands need to be found that result from basic strategy.

Let F be a card with a rank of 5 and T be a card with a rank of 10. Since play of the game involves the dealer exposing only one of his cards, let D be the “down card.” Suppose a hand is dealt, where the player is given two tens and the dealer is has one exposed ten. This is represented as (T, T) vs. (T, D), where the players hand is always listed first. The player takes no action (he stands on a total of twenty). The dealer is forced to act and he begins by exposing his down card. In this case, it is a ten. Therefore, the hand ends in (T, T) vs. (T, T), which is a push.

Determination of all N configurations requires an analysis of all possible hands dealt from (20, 32) that are played with basic strategy. For the first case, consider (T, T) vs. (T, D). It was previously mentioned that the player will stand. There are a total of three outcomes, the first of which is the dealer’s down card being a ten resulting in a push. The second outcome would be for the dealer to have a five. He draws a five for a total of 20 and a push. The third outcome is for the dealer's down card to be a five and for him to draw a ten. This would result in a bust and a win for the player.

Table A.1: Player (T, T) vs. Dealer (T, D).

Player's Cards			Dealer's Cards				Player's	
1st	2nd	3rd	Up	Down	3rd	4th	Action	Result
10	10		10	5	5		Stand	Push
10	10		10	5	10		Stand	Win
10	10		10	10			Stand	Push

The next case is (T, F) vs. (T, D). The player will stand on 15 or higher, forcing the dealer to act. This case is given in Table A.2.

Table A.2: Player (T, F) vs. Dealer (T, D).

Player's Cards			Dealer's Cards				Player's	
1st	2nd	3rd	Up	Down	3rd	4th	Action	Result
10	5		10	5	5		Stand	Lose
10	5		10	5	10		Stand	Win
10	5		10	10			Stand	Lose

Similar to Table A.2 is Table A.3. However, instead of the player holding (T, F), he is holding (F, T).

Table A.3: Player (F, T) vs. Dealer (T, D).

Player's Cards			Dealer's Cards				Player's	
1st	2nd	3rd	Up	Down	3rd	4th	Action	Result
5	10		10	5	5		Stand	Lose
5	10		10	5	10		Stand	Win
5	10		10	10			Stand	Lose

Tables A.4 thru A.6 are the remaining cases where the player takes no action. That is, the player has 15 or more and he stands on each hand.

Table A.4: Player (T, T) vs. Dealer (F, D).

Player's Cards			Dealer's Cards				Player's	
1st	2nd	3rd	Up	Down	3rd	4th	Action	Result
10	10		5	5	5	5	Stand	Push
10	10		5	5	5	10	Stand	Win
10	10		5	5	10		Stand	Push
10	10		5	10	5		Stand	Push
10	10		5	10	10		Stand	Win

Table A.5: Player (T, F) vs. Dealer (F, D).

Player's Cards			Dealer's Cards				Player's	
1st	2nd	3rd	Up	Down	3rd	4th	Action	Result
10	5		5	5	5	5	Stand	Lose
10	5		5	5	5	10	Stand	Win
10	5		5	5	10		Stand	Lose
10	5		5	10	5		Stand	Lose
10	5		5	10	10		Stand	Win

Table A.6: Player (F, T) vs. Dealer (F, D).

Player's Cards			Dealer's Cards				Player's	
1st	2nd	3rd	Up	Down	3rd	4th	Action	Result
5	10		5	5	5	5	Stand	Lose
5	10		5	5	5	10	Stand	Win
5	10		5	5	10		Stand	Lose
5	10		5	10	5		Stand	Lose
5	10		5	10	10		Stand	Win

Table A.7 covers the cases of (F, F) vs. (T, D). Basic strategy guides the player to hit a total of ten against an exposed ten. Clearly the player will only hit once: he will stand on a total of 15 or 20. In Table A.8, (F, F) vs. (F, D), the player will double-down (DD) his total of ten against a dealer five.

Table A.7: Player (F, F) vs. Dealer (T, D).

Player's Cards			Dealer's Cards				Player's	
1st	2nd	3rd	Up	Down	3rd	4th	Action	Result
5	5	5	10	5	5		Hit	Lose
5	5	5	10	5	10		Hit	Win
5	5	5	10	10			Hit	Lose
5	5	10	10	5	5		Hit	Push
5	5	10	10	5	10		Hit	Win
5	5	10	10	10			Hit	Push

Table A.8: Player (F, F) vs. Dealer (F, D).

Player's Cards			Dealer's Cards				Player's	
1st	2nd	3rd	Up	Down	3rd	4th	Action	Result
5	5	5	5	5	5	5	DD	Lose
5	5	5	5	5	5	10	DD	Win
5	5	5	5	5	10		DD	Lose
5	5	5	5	10	5		DD	Lose
5	5	5	5	10	10		DD	Win
5	5	10	5	5	5	5	DD	Push
5	5	10	5	5	5	10	DD	Win
5	5	10	5	5	10		DD	Push
5	5	10	5	10	5		DD	Push
5	5	10	5	10	10		DD	Win

Tables A.1 to A.8 result in a total of 40 unique hands that can be played beginning from (20, 32). Table A.9 combines the information from Tables A.1 to A.8 into a form that shows the beginning and ending distribution of cards, the number of cards removed, and the probability of reaching each ending distribution.

Table A.9: Probability of transitioning from (20, 32) to (20 – k, 32 – n).

Start Distribution: # Fives	# Tens	End Distribution: # Fives	# Tens	Cards Played: # Fives	# Tens	Likelihood: Frequency	P{Distribution}
20	32	13	32	7	0	1	0.00058
20	32	14	31	6	1	2	0.00265
20	32	15	30	5	2	1	0.00274
20	32	15	31	5	1	5	0.02031
20	32	16	30	4	2	8	0.06295
20	32	17	29	3	3	3	0.04166
20	32	17	30	3	2	7	0.15229
20	32	18	29	2	3	8	0.29009
20	32	19	28	1	4	2	0.11069
20	32	19	29	1	3	2	0.18321
20	32	20	28	0	4	1	0.13283

At the end of each hand is a new distribution of fives and tens. This is summarized in Table A.9, under the heading "End Distribution." The "Cards Played" shows the number of fives and tens removed from the deck. "Frequency" is a count of how many ways the End Distribution can be reached and P\{Distribution\} is the probability that (20, 32) will transition to (20 – k, 32 – n).

For example, the hand (T, T) vs. (T, T) means that four tens and zero fives were played out of the deck. Thus, the distribution of the deck moves from (20, 32) to (20, 28). This happens exactly one time. Thus, the probability of transitioning from (20, 32) to (20, 28) is:

$$P\{(20,28)\,|\,(20,32)\} = \left(\frac{32}{52}\right)\left(\frac{31}{51}\right)\left(\frac{30}{50}\right)\left(\frac{29}{49}\right)\times 1 = 0.133 \qquad \text{(A.1)}$$

The multiplication by one is because there is exactly one way to remove four tens from the deck.

For a transition from (20, 32) to (19, 29), the probability is:

$$P\{(19,29)\mid(20,32)\}=\left(\frac{20}{52}\right)\left(\frac{32}{51}\right)\left(\frac{31}{50}\right)\left(\frac{30}{49}\right)\times 2=0.183 \qquad \text{(A.2)}$$

Note that this is different from asking how many ways are there to remove 1 five and 3 tens from the deck. In this case, Equation (A.2) would be multiplied by 4 instead of 2. The transition from (20, 32) to (19, 29) occurs when one five and three tens have been removed from the deck. Only two possible hands satisfy this: player (F, T) vs. dealer (T, T) and player (T, F) vs. dealer (T, T) because basic strategy guides the player to stand on 15. On the other hand, were the player holding (T, T), that would mean the dealer was holding (T, F) or (F, T). Due to the structure of the game, the dealer does not stand on a total of fifteen: he must hit. The dealer will either draw a five ending the hand at (18, 29) or he will draw a ten ending the hand at (19, 28). This is captured in Figure A.9 under "Frequency."

Finally, summation of the column P\{Distribution\} equals one. Thus the forty unique hands end in one of eleven final distributions.