2019

ESTIMATION OF VOTER ARRIVAL RATES USING ELECTRONIC POLL BOOK TRANSACTION LOGS

James Houghton
University of Rhode Island, james_houghton@uri.edu

ESTIMATION OF VOTER ARRIVAL RATES

USING ELECTRONIC POLL BOOK TRANSACTION LOGS

BY

JAMES HOUGHTON

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

SYSTEMS ENGINEERING

UNIVERSITY OF RHODE ISLAND

2019

MASTER OF SCIENCE IN SYSTEMS ENGINEERING

OF

JAMES HOUGHTON

APPROVED:

Thesis Committee:

Major Professor Gretchen Macht

Thomas Wettergren

Jing Wu

Nasser H. Zawia
DEAN OF THE GRADUATE SCHOOL

UNIVERSITY OF RHODE ISLAND
2019

ABSTRACT

Election officials at the state and local levels begin operations and logistical planning several months before election day. Current research efforts focus on creating basic systems to facilitate the collection and synthesis of polling place data by election administrators and poll workers. Practically all the current methods involve the manual collection of this data, and then some aggregated form is utilized in decision-making processes. The research contributes to the voting-systems literature in two ways. First, it broadens the scope of knowledge about check-in processing time variation both within and between precincts. Secondly, it proposes a methodology for using the EPB transaction logs to estimate arrival rates using a Hidden Markov Model.

Check-In processing time observations are collected through time studies during the 2018 Midterm elections at seven precincts throughout Rhode Island. An analysis of check-in observations revealed that processing times are reasonably similar both between and within precincts. Check-In observations are then used to model a stochastic process time distribution for four precincts in Providence, Rhode Island. The process models are combined with electronic poll book transaction logs to simulate voter arrival times. The count of simulated arrivals over discrete 15minute intervals are used to populate an observation sequence. Multiple observation sequences are used to compute parameter estimates for a Discrete-time Poisson Hidden Markov Model (dt-PHMM).

A dt-PHMM is constructed with three, four, and five hidden states for each of the four Providence Precincts. At least one dt-PHMM model was successfully able to estimate arrival rates for all four precincts. The most appropriate size for the hidden state-space varied between precincts. The strengths and weakness of the three, four, and five-state models are discussed for each precinct.

ACKNOWLEDGMENTS

First and foremost, I would like to thank my major professor Dr. Gretchen Macht. Her support and encouragement over the last two years have helped me grow as an academic and a young professional. Her compassion, teamwork, and innovation have helped me to learn new skills and explore new domains. I would also like to thank Dr. Thomas Wettergren for challenging me to think critically and take intellectual leaps in my research. Thank you also to Dr. Jing Wu and Dr. Noah Daniels for your outside perspectives and guidance as I completed my thesis research. It has helped me to gain a deeper understanding of my work. Lastly, I would like to thank the Dean of the College of Engineering, Dr. Raymond Wright, and the Democracy Fund for allowing me to work on the RI VOTES project and their support along the way.

TABLE OF CONTENTS

LIST OF TABLES

TABLE PAGE

LIST OF FIGURES

FIGURE PAGE

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

Election officials at the state and local levels begin operations and logistical planning several months before election day. It is common for Election Officials to make resource allocation decisions using "Rules of Thumb." Their planning heuristics are frequently based on their past decisions rather than quantitative methods (Stewart III, 2015). Election administrators often lack the necessary data to effectively measure election performance and identify operational inefficacies (Spencer & Markovits, 2010).

Furthermore, simulation studies have shown that voter wait times are extremely sensitive to changes in voter turnout and expected processing time (Edelstein & Edelstein, 2010). The arrival behavior of voters on Election Day can significantly impact the resources (e.g., number of poll workers, voting booths, and ballot scanners) required to manage queues at a polling place. It is especially important that election officials can adequately estimate the capacity needs for each precinct as these new technologies can be expensive and are often a scarce resource (Yang, Kelton, Fry, & Allen, 2013).

Recent literature relating to resource allocation and voter wait times are based on observational (i.e., time studies and survey response) or queuing theory and simulation models (Herron & Smith, 2015). Time studies are useful for measuring processing times and voter arrivals but can labor-intensive and require a fair deal of planning. (Fortier, Stewart III, Pettigrew, Weil, & Harper, 2018). Survey-based research allows

for arrival and wait time information to be collected at the national level but often lack the specificity to perform analyses at the municipal or precinct levels (Stewart III, 2015). Queuing simulations models and operations management approaches provide trackable methods for effective capacity planning and resource allocation. The results, however, are based on broad generalizations about voter arrival behaviors derived from synthetic data or historical case studies. Application of these models requires precinct specific knowledge about the arrival behavior and processing times per jurisdiction per election.

Various researchers have been working on designing and implementing simple data collection programs to provide actionable data to about voter arrivals and queue lengths and processing times. A report by the Bipartisan Policy Center (BCP) posits that information about voter arrivals and line lengths must be collected regularly at every single polling place in a given jurisdiction (Fortier, Stewart III, Pettigrew, Weil, & Harper, 2018). Current research efforts focus on creating basic systems to facilitate the collection and synthesis of polling place data by election administrators and poll workers.

1.2 RESEARCH GOALS

Practically all the current methods involve the manual collection of this data and then some aggregated form to be implemented in decision-making processes. Perhaps a more efficient, scalable, and sustainable method can be developed using timestamped voter check-in information captured automatically by the electronic poll book (EPB) systems. Electronic Poll books exist on laptops or tablets that are directly connected to the Statewide Voter Registration System. In a report to the Wisconsin Government Accountability Board, Michael Hass discusses several new improvements introduced by EPB systems. Firstly, they eliminate the need the for alphabetically divided poll rosters to provide multiple check-in stations allowing voters to check-in at the first available stations. EPBs are also able to look up voters automatically by scanning the barcode on their ID card.

Furthermore, the EPB identifies if voters are at the wrong location and share the address of the correct location to their cell phone via text. Lastly, EPBs can upload election-day registrations to the Statewide Voter Registration System automatically instead of entering it manually - which can be time-consuming and prone to human error. This study's research aims to understand the characteristics of the integrated EPB check-in process to leverage its transaction logs to obtain critical insights about arrival behavior at polling places for a more immediate feedback loop and process enhancement for overall improved decision making for the future of elections.

Therefore, the research questions for this work are:

1. What are the process time characteristics at the check-in station using EPB's?
2. How does the processing time vary between poll workers in a single polling place? Between different polling places?
3. Can the EPB check-in timestamps be used as a proxy for actual arrival times? If so, to what extent?
4. Can the EPB check-in timestamps be used in a Hidden Markov Model (HMM) to reveal voter arrival patterns?

The remainder of this thesis proceeds as follows. Chapter 2 begins by reviewing current methods for collecting and analyzing voter arrival data. Potential strengths and weaknesses of these methodologies are then discussed. The third chapter outlines the methodology and implementations used to address the research questions proposed in this study. The methodology section begins with a basic overview of the procedures used to collect and clean various elections related data sets from polling places in Rhode Island (RI) during the 2018 U.S. Midterm elections. This will include processing time data from times studies as well as electronic records provided by the Rhode Island Board of Elections (RI BOE). The second half of Chapter 3 focuses on the fundamental terms and concepts that will be used to estimate arrival rates at four different precincts. Chapter 4 first reports the analysis results of check-in time study data then continues with a detailed description of the HMM implementation and results for each of the four test precincts. The fifth and final chapter accesses the adequacy of the proposed methodology and concludes the proposed research questions.

CHAPTER 2

REVIEW OF LITERATURE

This chapter provides an overview of past case studies as well as current methods for collecting and analyzing elections data with a focus on voter arrival behavior.

2.1 CASE STUDIES

The use of queuing theory and simulation optimization models to examine resource allocation decisions within the voting systems domain is extremely limited. Allen and Bernshteyn (2006) use a basic queuing theory model to predict average wait times and voting-machine requirements. Allen and Bernshteyn (2006) use data from Franklin County, Ohio, during the 2004 presidential election to create a machine-allocation algorithm to minimize wait times and maximize efficiency.

A heuristic approach for mitigating wait times using a "Queue Stop Rule" is proposed by Edelstein (2006) to determine the minimum number of parallel servers needed at each station to prevent voter wait times from exceeding a prespecified value. The time to vote, T_{Vote} is given by Equation 1 (T_{Day} is the amount of time the polling place is open, and NV_{vs} is the total number of voters).

$$T_{Vote} \leq \frac{1}{2}\left(\frac{T_{Day}}{NV_{vs}}\right) \qquad (1)$$

Edelstein & Edelstein (2010) expands upon this work by include variable arrival rates using a Non-homogenous Poisson Process. A numerical example offered by the authors assumes three high-intensity periods between 7:00 AM-9:00 AM, 12:00 PM-2:00 PM, and 5:00 PM-8:00 PM with 10% of total voters arriving each hour. Arrivals occurred at 5% per hour at all other times. The baseline process times and arrival patterns in these studies are based on the observations in (Dow, 2007). The authors

report that the Queue Stop Rule output very sensitive to process time and arrival rates and conclude with a discussion of specific considerations for designing efficient voting systems.

Two different heuristic approaches for making fair and effective resource allocation decisions are explored in (Yang, Fry, & Kelton, 2009). Voting times are based on a mock election using the 2006 gubernatorial election ballot. The Voter Experience Survey (Feldman & Belcher, 2005) from Franklin County, Ohio, is used to model the arrival distribution shown in Table 1.

Time Interval	Arrival Percentage
Before 8:00 AM	20.61
8:00 AM-11:00 AM	27.34
11:00 AM-3:00PM	24.05
3:00 PM-5:00 PM	13.26
After 5:00 PM	13.87

Table 1: Arrival Distribution of Voters in Yang et al. 2009

A proposed "Greedy Improvement Algorithm" (GIP) heuristic aims to minimize the average absolute difference of expected waiting times across all precincts. The authors also implement a Utilization Equalization heuristic approach that balance the resource utilization levels across all polling places. (Yang, Kelton, Fry, & Allen, 2013) builds upon this work by formalizing an optimization model and exploring various objective functions that minimize the maximum average waiting times for a given set of precincts. Their work also discusses how existing Service-Operations Management techniques (capacity & demand management) apply to voting systems.

Herron & Smith (2015) define a generalizable procedure for collecting data on voter arrivals and processing times using web-based applications. The data collection procedure is implemented during the 2014 Midterm election for a single polling place

in Hanover, New Hampshire. A simulation model is constructed based on this data and used to estimate voter wait times under 36 different scenarios These scenarios explore two arrival patterns and various combinations of resource allocation levels for check-in stations, voting booths, and ballot scanners.

2.2 ARRIVAL RATES

The work of Spencer and Markovits (2010) develops a systematic data collection method that breaks down the voting process into three fundamental steps that can be universally applied regardless of local rules. This method is used in a pilot study to collect arrival and processing time observations during the 2008 presidential primary for 30 polling stations across three counties California. The data is collected by stationing volunteers inside the polling place. Volunteers record the number of voters arriving in 10-minute intervals between 7:00 AM and 8:00 PM. Service times were recording by recording a timestamp for every fifth voter as they started and finished each operation (check-in, ballot marking, ballot scanning). These observations are used in a basic queuing model to predict line lengths and identify potential bottlenecks in the process. The check-in and ballot marking times were found to be relatively constant despite changes in the arrival rate of voters. The author posits that the arrival rate of voters may be predictable based on a "double-hump" pattern of increased arrival rates in the early morning and late afternoon.

Survey research from the 2012 and 2016 General Elections by (Stewart III, 2015) found that peak arrival rates generally occurred early in the morning, steadily decline throughout the afternoon and then increase slightly in the evening as voters leave work. This supports the "double-hump" pattern suggested by (Spencer &

Markovits, 2010). However, Stewart also reported significant variation at the county, and state levels (2015). These findings were based on responses to the Survey of the Performance of American Elections (SPAE) and the Cooperative Congressional Election Study (CCES).

2.3 PRACTICAL TOOLS AND APPLICATIONS

The BPC and MIT collaborated to develop a survey-based protocol that would empower poll workers to collect actionable data about wait times and line dynamics. A "Line Length survey" collected hourly counts of line length at 88 different precincts in 2016 General elections. Average wait times were reported for each precinct using Little's Law given by Equation 2 The average arrival rate was calculated by dividing the total number of voters by the total time, in minutes, the polling place was open.

$$Average\ Wait\ Time = \frac{Average\ Line\ Length}{Average\ Arrival\ Rate} \qquad (2)$$

An additional report was included for municipalities where hourly check-in counts were available through their EPB transaction logs. Figure 1 illustrates details about the data included in the additional report using a line graph.

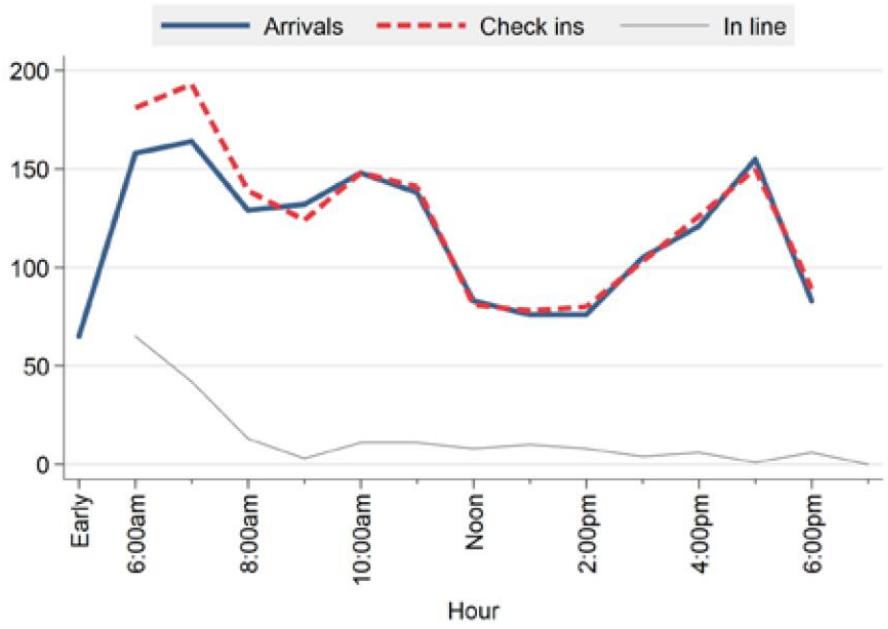

Figure 1: Hourly Arrivals, Check-ins, and Number Standing in Line Source: BCP Voting Lines Project

The authors acknowledge that Littles law is only valid over the long run and that the hourly line count is not always an accurate representation of the actual average line length. The authors argue that their current protocol strikes a workable balance between precision and cost — their survey designed for simplicity so that it can be implemented universally without increasing staffing requirements.

The Voter Technology Project (VTP), a collaborative effort between CalTech and the Massachusetts Institute of Technology (MIT), significantly contributes to the polling place resource management and election planning literature. A 2015 publication written by Stuart, titled *Managing Polling Place Recourses,* describes the fundamental concept of Queueing Theory and explains the potential benefits for resource allocation decision making. This document provides simple data collection procedures for measuring arrival rates and process times along with instructions for using resource allocation tools available on the VTP website. The Line Optimization

and Poll Worker Management tool can be used to estimate of the number of check-in stations or voting booths required to attain a desired average wait time or service level at a polling place. A screenshot of this tool is provided in Appendix A. This tool is simple, straightforward, and well documented.

A second tool by the VTP, called the Line Optimization tool shows average expected wait times throughout the day. The Line Optimization tool accounts for potential bottlenecks at the check-in station and voting booths. The Line Optimization tool also allows users to choose between a smooth arrival pattern and an early morning peak. The arrival patterns for election day are based on data collected by in a study by Fortier, Stewart, Pettigrew, Weil, and Harper (2018) However, significant variation in arrival patterns between polling places at various municipal levels (Stewart III, 2015). The output of this tool may not be representative of any specific precinct unless it is known to have a similar arrival distribution.

The tools provided by the VTP and others are certainly a step in the right direction. However, a more efficient, scalable method for approximating voter arrival behavior is needed. The new electronic poll book systems automatically capture timestamped voter check-in information. Their transaction logs may provide useful insights about voter arrival behavior. This study's research contributes to the voting-systems literature in two ways. First, it broadens the scope of knowledge about check-in processing time variation both within and between precincts. Secondly, it proposes a methodology for using the EPB transaction logs to estimate arrival rates using a Hidden Markov Model.

CHAPTER 3

METHODOLOGY

The following chapter describes the methodology and procedures used to address the proposed research questions. The first section describes the time study procedures used to collect processing time data at RI polling places during the U.S. Midterm election in 2018. The methods to clean and validate the check-in processing time observations are discussed, followed by a comparative analysis. The next section introduces the EPB log files, data validation processes, and comparative analysis methods. The HMM implementation procedure for estimating voter arrival rates is discussed in the final sections of this chapter.

3.1 POLLING PLACE TIME STUDIES

A series of time studies were performed during the 2018 U.S. Midterm Elections at seven Polling places throughout Rhode Island. Simple timers were created in Microsoft Excel using Visual Basic for Application (VBA) forms to ensure timestamps were precise and consistently formatted. Separate timers were used to record observations for each check-in stations so that processing times could be compared on an individual basis. Students enrolled in the Human Factors and Ergonomics class (ISE/PSY 420) at URI were trained on how to use the VBA timers before participating in the time study. A timestamp was recorded for the Check-in start time when a voter engaged with poll workers at an individual station. A second timestamp was recorded as the end time when a voter accepted their ballot and exited the station. The complete Data Collection Instructions document and a preview of the timing tools are provided in Appendix B. Students were grouped in teams of four and

assigned to collect data from 7:00 AM-11:00 AM at three different polling locations across the State. The specific timeframe was chosen to maximize the number of observations collected. The fifth team with two Graduate Research Assistants and an ISE professor collected data from precincts across Providence, RI between 7:00 AM and 7:30 PM. This team also collected anecdotal evidence about voter arrival behaviors from election officials and local poll workers. Individual Excel files for each precinct were saved using a standardized naming convention which included the observers last name and the station observed.

3.1.1 DATA PROCESSING AND ANALYSIS

The time studies data files were collected from all participants and organized into folders by station type. The check-in observations for all locations were then consolidated onto a CSV file with the columns labeled Precinct Number, Station Number, Start Time, End Time, Observer Last Name. The CSV file was then imported as a Data Frame using the Pandas package in Python. An additional column called "Seconds" computed the check-in times by subtracting End Time and Start Time columns.

3.1.2 COMPARITIVE ANALYSIS

The process time observations for individual poll pads within each precinct are compared using a Kruskal-Wallis test. This non-parametric test was chosen due to the varying sample size between poll pads (Kruskal & Wallis, 1952). The null hypothesis that the population median is equal for all test groups is tested using a P-value of 0.05. Post-hoc comparisons are performed when the null hypothesis is rejected to identify which individual group(s) are different following (Dinno, 2015). The Kruskal-Wallis

test is also used to compare aggregated precinct data. This testing only includes precincts where all check-in station observations were similar. The results of comparative tests within precincts will determine if a single processing time distribution can be used to represent all Check-In stations. The comparative analysis between precincts is used to assess if a generalized Check-in process model could be used in cases where precinct specific data is not available.

3.1.3 DEFINING CHECK-IN PROCESS MODELS

A stochastic process model for each precinct is used to generate observation sequences used to train Hidden Markov models for precincts 1-4. The check-in observations were fit to a variety of statistical distributions using the Fitter package in Python. This package uses SciPy's fit method is used to extract Maximized Likelihood Estimates (MLE) for the parameters each distribution tested. The sum-square error (SSE) is used to report the goodness-of-fit for each distribution. The Lognormal distribution performed well for the four providence precincts. The SciPy package defines the MLE parameters for the Lognormal distribution as *Shape* and *Scale*. The *Shape* parameter is equal to the natural log of the observed standard deviation (std. dev). The *Scale* computed based on the observed mean. Parameter estimates are defined for each precinct in Table 2.

Precinct	Scale	Mean	Shape	Std. Dev	SSE
1	51.1097	56.1689	0.0434	25.6030	0.00368
2	41.5438	44.7884	0.388	18.0436	0.00383
3	50.9032	55.8517	0.431	25.2186	0.00884
4	55.0285	63.3564	0.530895	36.1507	0.00616

Table 2: Stochastic Process Model Parameter Estimates

3.2 ELECTRONIC POLL PAD DATA

The EPB transaction logs record the following information for every voter on Election day: Sequential ID number, Election Name, Timestamp, Poll Pad Name, and Precinct Number upon completion of the check-in process. The transaction logs from the EPB's used in Rhode Island polling places during the 2018 elections are the primary source of raw data used to approximate arrival rates. These files were provided by the RI BOE as a Microsoft Excel file using the ".xlsx" format. A data validation script was created to identify and correct any irregularities that may have occurred while transferring the data. The first function in this script tested the Timestamp column to ensure all values were displayed in the correct time zone using the "MM:DD:hh:mm: ss" format. The second function is used to ensure only one precinct number and location name is recorded for each EPB. A third function is used to create a new column with the Timestamp corresponding to the previous check-in on that device. This column will be referenced when simulating voter arrival times.

3.2.1 PRESCREENING PRECINCT EPB DATA

The timestamps in the transaction logs record the time when a voter completes the check-in process but does not indicate the starting time nor the exact time of arrival. This chapter defines a procedure for using the process models defined in 3.1.3 to estimate arrival rates over discrete, 15-minute intervals. First, Pseudo-start times are computed by subtracting the average observed check-in time from each EPB timestamp. These values are used as a proxy for arrival times under the assumption that queue formation is minimal, and any delay that occurs between the time a voter arrives and begins the check-in process is negligible. To test this, a deterministic

throughput capacity is estimated based on the average of time studies observations for each precinct. The maximum and 75% throughput capacities are plotted against the pseudo-arrival counts to graphically access if the throughput capacity was sufficient throughout the day.

Next, the proxy arrival times tested for conformance to a non-homogenous Poisson Process. The time between successive arrivals must be exponentially distributed with a stationary mean when separated into independent time blocks. A Log Kolmogorov-Smirnov test (Brown, et al., 2006) is applied the proxy arrival times over 15-minute intervals. First, the data be transformed using Equation 3:

$$R_{ij} = (J(i) + 1 - j\left[-ln\left(\frac{L - T_{ij}}{L - T_{i,j-1}}\right)\right] \qquad (3)$$

Where T_{ij} is the *j*-th ordered arrival time in the *i*-th block, J(*i*) is the total observations in the *i*-th block, and L is the time length of each block.

Next, a Kolmogorov-Smirnov test is used to test the null hypothesis:

H_0: $\{R_{ij}\}$ are independent, standard exponential variables

A False Detection Rate procedure (Benjamini & Hochberg, 1995) is applied to the P-values for all intervals to correct false positives. If the null hypothesis fails to be rejected, then it is concluded that arrival observations in each interval are a Poisson Process.

3.3 POISSON HIDDEN MARKOV MODEL

Non-homogenous Poisson processes have been used to represent voter arrival rates in previous voting systems research. Edelstein & Edelstein (2009) and Yang et al. (2009; 2013) rely on rate time tables that assume the rate to be constant over intervals ranging for two to four hours. Herron & Smith (2015) use smaller, one-hour intervals. The level of variability illustrated in the arrival count plots throughout Chapter 4.2 show these estimates to be gross overgeneralizations.

This research proposes a probabilistic approach to model the evolution of arrival rates of individual precincts using a special case of Hidden Markov Models (HMM) called a Discrete-time Poisson Hidden Markov Model (dt-PHMM). An HMM is a bivariate market chain that combines an observable time stochastic process $\{O_t\}$ with a hidden Markov chain $\{C_t\}$ with states that cannot directly be observed.

HMMs are useful for temporal pattern recognition and are especially known for their use cases in speech recognition (Rabiner, 1989). In these use cases, the hidden Markov chain is predefined using semantically meaningful states. In the case of Poisson HMMs, the modeler seeks to define the hidden Markov Chain in with states corresponding to meaning rate classes. Two popular use-cases in HMM literature that use PHMMs to analyze count data in discrete time include Leroux & Puterman (1992) and Scott (2001). Leroux and Puterman (1992) construct a PHMM to monitor Fetal lamb activity with hidden states that signify periods where the lamb was *inactive, somewhat active*, or *very active*. Scott (2001) constructs a PHMM with a binary hidden state to predict internet network intrusion based on mouse-click activity.

A discrete-time PHMM was also used by Paroli, Redaelli, & Spezia (2002) for over-dispersed insurance counts.

For Poisson Hidden Markov models defined for the purpose of this study, the unobservable sequence $\{C_t\}$ exists within the finite state-space $S_c = \{1,2\ldots m\}$. Each O_t in the observed sequence, $\{O_t\}$ is conditioned on the contemporary state of C_t (Paroli, Redaelli, & Spezia, 2002). For any time t, where C_t is in state i ($i \in S_c$), the conditional distribution of O_t is a Poisson random variable with rate parameter λ_i (Paroli, Redaelli, & Spezia, 2002). The An m-length vector δ is the initial state distribution of C_t at t =1. The parameter A denotes an m x m matrix where θ_{ij} is the transition probability from state i, at time t-1, to state j at time t (for any state i,j, and for any time t). Additional elements used to characterize the model are defined as follows:

- O_t = Number of arrivals observed in time interval *t*,
- Q = $q_1, q_2, \ldots q_T$ be the state sequence where q_T is in state i at time *t*.
- $\pi_{O,i} = P(O_t = O \mid C_t = i)$, the state-dependent probabilities

The state dependant probabilities are computed using Equation 4 given by Paroli, Redaelli, & Spezia (2002):

$$\pi_{O_t,i_t} = \mathrm{e}^{-\lambda_{i_t}} \frac{\lambda_{i_t}^{O_t}}{O_t!} \qquad (4)$$

Implementation of a dt-PHMM can be broken down into three general steps. First, the *Forward-Backward algorithm* (Rabiner & Juang, 1986) is used to compute $P(O|\phi)$, the probability that observed sequence $\{O_t\}$ will occur given the initial parameter estimate for the model: $\phi = [\delta, \theta, \lambda]$ (5). In the next step, the *Baum-Welch*

algorithm (Baum, Petrie, & Weiss, 1970) maximizes the probability of observing sequence $\{O_t\}$ by iteratively adjusting parameter estimates for $[\delta, \theta, \lambda]$. Finally, the *Viterbi algorithm* is applied to find the hidden state sequence, S*, that is most likely to generate O, the observed sequence (Viterbi, 1967). The Forward-Backward and Baum-Welch algorithms are implemented using the procedures and equations given by Paroli, Redaelli, & Spezia (2002, p464-466)[1]. Implementation of the the Logrithmic Viterbi algorithm follows Tiberiu & Harrison (2013, p77-85)[2]

3.3.1 FORWARD-BACKWARD ALGORITHM

The Forward-Backward algorithm computes the probability of observing a given sequence ($P(O|\phi)$, in terms of forward and backward variables denoted by α and β.

Forward variable:

$$\alpha_t(i) = P(O_1, O_2 \dots, O_t, q_t = S_i \mid \phi) \qquad (6)$$

Where i = 1, 2,…m , t = 1,2,…T, and S_i is the state at time t

1. Initialization:

$$\alpha_1(i) = \delta_i \pi_{O_1,i} \ for\ 1 \le i \le m \qquad (7)$$

2. Proceeding inductively:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{m} \alpha_t(i)\theta_{ij}\right] \pi_{O_{t+1},j} \qquad (8)$$

$$for\ 1 \le t \le T-1\ , 1 \le j \le m\ .$$

Where $\alpha_t(i)\theta_{ij}$ is the joint event probability of observing $O_1, O_2 \dots, O_t$ then transitioning from state S_i at time t to state S_j at time $t + 1$.

[1] The parameter notation is slightly different. The parameters X,Y, and γ correspond to the parameters C, O, and θ used in this study.

[2] The author's parameters $\delta, a, b,$ and π correspond to the parameters denoted by $\zeta, \theta, \pi,$ and δ in this study

Backwards Variable:

$$\beta_t(i) = P(\mathrm{O}_{\mathrm{t+1}}, \mathrm{O}_{\mathrm{t+2}} \ldots, \mathrm{O}_{\mathrm{T}} |\, q_t = S_i, \phi) \tag{9}$$

1. Initialization:

$$\beta_T(i) = 1, \quad For\ 1 \le i \le m \tag{10}$$

2. Proceeding Inductively:

$$\beta_t(i) = \sum_{j=1}^{m} \theta_{ij}\, \pi_{O_{t+1},j}\, \beta_{t+1}(j) \tag{11}$$

$$for\ t = T-1, T-2, \ldots, 1, \qquad 1 \le i \le m\,.$$

Finally, the observation probability is given by Equation 16:

$$P(O|\phi) = \sum_{i=1}^{m} \alpha_t(i)\beta_t(i),\ \forall\ \mathrm{t} \tag{12}$$

3.3.2 BAUM-WELCH ALGORITHM

The Forward and Backward probabilities are used by the Baum-Welch algorithm to find the Maximized Likelihood Estimator of ϕ . $[\delta, \theta, \lambda]$. A two-step Expectation-Maximization (EM) procedure iteratively adjusts parameter estimates as defined by (Paroli, Redaelli, & Spezia, 2002). The E step computes Forward and Backward probabilities according to Equations 12 & 13, respectively. Next, the auxiliary function for the $(k+1)^{th}$ iteration is evaluated based on using Equation 13a as follows:

$$Q(\phi; \phi^k) = E_{\phi^k}(\ln L_T(\phi) |\, \mathrm{O}) \tag{13}$$

$$= \left(\sum_{i \in S_c} \frac{\alpha_1^k(i)\beta_1^k(i)}{\sum_{i \in S_c} \alpha_t^k(i)\beta_t^k(i)} \ln \delta_i\right) + \left(\sum_{i \in S_c} \sum_{j \in S_c} \frac{\sum_{t=1}^{T-i} \alpha_t^k(i)\, \theta_{i,j}^k \pi_{O_{t+1},j}^k \beta_{t+1}^k(j)}{\sum_{i \in S_c} \alpha_t^k(i)\beta_t^k(i)} \ln \theta_{i,j}\right) +$$

$$+ \left(\frac{\sum_{t=1}^{T} \alpha_t^k(i)\beta_t^k(i)}{\sum_{i \in S_c} \alpha_t^k(i)\beta_t^k(i)} \ln \pi_{O_t,i}^k\right) \tag{13a}$$

The M-step $(k+1)^{th}$ iteration seeks to maximize auxiliary function, such that $Q(\phi^{k+1};\phi^{k}) \geq Q(\phi;\phi^{k})$. Maximum Likelihood Estimates are obtained for $\hat{\theta}$ and $\hat{\lambda}$ applying Equations 14 & 15, respectively.

$$\theta_{i,j}^{k+1} = \frac{\sum_{t=1}^{T-i} \alpha_t^k(i)\ \ \theta_{i,j}^k\ \ \pi_{O_{t+1},j}^k\ \ \beta_{t+1}^k(j)}{\sum_{t=1}^{T} \alpha_t^k(i)\ \ \beta_t^k(i)} \tag{14}$$

$$\lambda_i^{k+1} = \frac{\sum_{t=1}^{T} \alpha_t^k(i)\ \ \beta_t^k(i)\ \ O_t}{\sum_{t=1}^{T} \alpha_t^k(i)\ \ \beta_t^k(i)} \tag{15}$$

The Baum-Welch algorithm repeats between the two EM steps until $(\ln L_T(\phi^{k+1})$ - $(\ln L_T(\phi^{k})$ converges to a difference less than 1.00 e-3.

3.3.3 VITERBI ALGORITHM

The Logarithmic Viterbi Algorithm is used to identify the most likely hidden state sequence, S* following the four-step procedure described by Tiberiu & Harrison (2013). The variable S_t*(i) is used to denote the path ending in S_i that maximizes log-likelihood for observations $O_1, O_2, \ldots, O_t$. The variable $\zeta(i)$ computes the log probability of generating observations $O_1, O_2, \ldots, O_t$, from path S_t*(i). The variable $\psi_t(i)$ is defined to track each *t* and *i* that has maximized the last $\zeta_t(i)$ (Tiberiu & Harrison, 2013). The Logarithmic Viterbi Algorithm proceeds as follows:

1. Initialization:

$$\zeta_1(i) = \ln(\delta_i b_i O_1)\ 1 \leq i \leq m \tag{16}$$

$$\psi_1(i) = 0 \tag{17}$$

2. Recursively compute values for variables for *j* = 1,2, …, m and *t* = 1,2, …,T-1:

$$\zeta_t(j) = max_{1\leq i\leq m}\left[\zeta_{t-1}(i) + \ln\theta_{ij}\right] + \ln(b_j(O_t)) \tag{18}$$

$$\psi_t(j) = argmax_{1\leq i\leq m}\left[\zeta_{t-1}(i) + \ln\theta_{ij}\right] \tag{19}$$

3. Termination:

$$P^* = max_{1 \le i \le m}[\zeta_T(i)] \quad (20)$$

$$S_T = argmax_{1 \le i \le m}[\zeta_T(i)] \quad (21)$$

4. Backtrack through the sequence as such:

$$S_t^* = \psi_{t+1}(i_{t+1}^*) \quad (22)$$

3.3.4 INITIALIZING MODEL PARAMETERS

The proxy arrival times from the Log-KS test are used to generate an array of arrival counts at 15-minute intervals between 7:00 AM and 8: 00 PM. The k-means algorithm is used group the arrival counts into m different clusters. The cluster centers are then used to define λ_0, the initial vector of rate parameters. Parameter values for the initial state and transition probability matrices are defined arbitrarily at first and then adjusted manually until the Baum Welch algorithm converges.

3.3.5 GENERATING OBSERVATION SEQUENCES

Observation sequences for each precinct are generated from EPB data using the stochastic process model. Arrival times are simulated for each check-in observation by subtracting a random variable from the process time distribution. For continuity, the simulated arrival time is replaced by the previously observed check-in time of that machine if the simulated time proceeds the EPB timestamp. The count of simulated arrivals over discrete 15minute intervals are used to populate an observation sequence $\{O_t\}$. Multiple observation sequences are used to train the model to provide more reliable estimates of model parameters (Rabiner, 1989).

3.3.6 EVALUATING MODEL FITNESS

A simple, theoretically correct method for estimating the most appropriate number of states has not yet been established (Rabiner, 1989). Three models are constructed for each precinct with the number of hidden states m = 3,4, and 5. Precinct models are compared quantitatively using two maximum penalized likelihood estimators following (Leroux & Puterman, 1992). The Bayesian Information Criterion (BIC) and Akaike Information Criterion (AIC) and computed using Equations 23 & 24, respectively.

$$BIC = \ln(\hat{L}) - \frac{1}{2}\ln(n) * k \quad (23)$$

$$AIC = \ln(\hat{L}) - k \quad (24)$$

Where: $\hat{L}$ is the maximized value of the likelihood function of the model ϕ. The total number of data points (sequence length x number of samples) is denoted by n. The variable k represents the number of parameters to be estimated under the model. The fitted values for $\hat{\lambda}$ and $\hat{\theta}$ are also taken into consideration when comparing the models for each precinct. Value in the matrix $\hat{\lambda}$ should represent a unique, semantically meaningful rate class that, at least vaguely, describes an arrival intensity for each state (i.e., high, moderate, low intensity). The transition probabilities in the $\hat{\theta}$ matrix need not be fully ergodic. However, model validity is rejected if $\hat{\theta}$ contains closed, or absorbing states.

CHAPTER 4

FINDINGS

This chapter begins by summarizing the check-on processing time data collected during the U.S. 2018 Rhode Island Midterm election. A comparative analysis of processing times is then performed for observations within and then between precincts. The second half of the chapter details the implementation and evaluation of Poisson Hidden Markov models constructed for four precincts in Providence, Rhode Island.

4.1 COMPARITIVE ANALYSIS OF CHECK-IN PROCESSING TIMES

Processing time observations were collected for a total of 25 check-in stations across seven different polling places. The number of check-in stations varies between precincts, as well the number of observations recorded for each station. Table 3 provides a general overview via descriptive statistics of processing time observations from the 2018 Midterms time study.

Precinct	Poll Pad	Count	Average	Standard Deviation
1	1_1	46	59.09	24.74
	1_2	28	59.89	25.18
	1_3	47	50.55	25.56
	1_4	56	57.13	28.39
2	2_1	40	42.40	16.98
	2_2	89	44.29	19.85
	2_3	73	48.36	21.68
	2_4	55	44.00	21.88
3	3_1	18	60.00	28.63
	3_2	26	54.19	23.78
	3_3	40	52.53	26.44
	3_4	27	61.07	31.33
4	4_2	16	61.63	47.41
	4_3	9	58.67	43.28
	4_4	16	70.94	42.29
5	5_1	89	56.69	19.01
	5_2	79	60.03	17.99
	5_3	94	45.65	17.75
	5_4	65	44.14	26.24
6	6_1	67	63.13	32.33
	6_2	58	59.83	22.38
	6_3	86	61.00	33.70
	6_4	68	49.57	18.58
7	7_1	17	87.24	29.48
	7_2	11	105.73	36.16

Table 3: Summary of Check-in Observations

4.1.1 PRECINCT LEVEL COMPARISONS

The observations from each precinct are compared using a Kruskal-Wallace test. Table 4 summarized the results for individual precincts.

Precinct	No. Stations	Total Observations	Test Statistic	P-Value
1	4	177	6.82	0.078
2	4	257	4.97	0.174
3	4	111	2.43	0.488
4	3	41	2.34	0.309
5	4	327	65.77	0.00*
6	4	279	12.57	0.006*
7	2	28	1.799	0.180

Table 4: Kruskal-Wallace test for Individual Precinct Observations

Post-hoc testing is performed for these precincts to determine which check-in station observations are different and identify subsets of similar observations for Precincts 5 and Precinct 6. The P-values from the Dunns-Bonferroni test, Table 5, indicate subgroups of similar data can be formed for Precinct 5 for the first and second then third and fourth check-in stations. Table 6 lists the P-values for the Dunns-Bonferroni test for Precinct 6 observations. The first, second and third stations are all similar. The fourth station is similar to station three but, significantly different from stations one and two.

Station	1	2	3	4
1	-1.00	1.00	1.82e-05	1.27e-07
2	1.00	-1.00	4.07e-08	2.20e-10
3	1.82e-05	4.08e-08	-1.00	9.96e-01
4	1.27e-07	2.20e-10	9.96e-01	-1.00

Table 5: Dunn-Bonferroni test for Subgroups within Precinct 5

Station	1	2	3	4
1	-1.00	1.00	1.00	9.88e-03
2	1.00	-1.00	1.00	1.99e-02
3	1.00	1.00	-1.00	1.83e-01
4	9.88e-03	1.99e-02	1.83e-01	-1.00

Table 6 Dunn-Bonferroni test for Subgroups within Precinct 6

4.1.2 COMPARING BETWEEN PRECINCTS

A comparative analysis is performed between each precinct to address the second research question posed in this study. This is performed using aggregated data from precincts, where all poll pads observations were found to be similar. Precincts with P-values less than 0.05 in the initial precinct level testing are excluded from this analysis (Precinct 5 and 6). The Kruskal-Wallace test was performed using aggregated data from precincts 1, 2, 3, 4, and 7 and confidently rejected the null hypothesis that all precincts were similar. P-values from a post-hoc analysis using a Dunns-Bonferonni test, Table 7, indicate that aggegate observations from Precinct 1, 3, and 4 are all similar to one another.Obervations from Precinct 2 and 7 were not similar to any other prcincts.

Precinct	1	2	3	4	7
1	-1.00	3.06e-07	0.788	0.697	0.02
2	3.06e-07	-1.00	3.4e-03	7.34e-03	1.86e-15
3	7.89e-01	3.84e-05	-1.00	0.584	2.08e-02
4	6.97e-01	7.39e-04	0.584	-1.00	3.16e-01
7	1.93e-02	1.86e-15	0.020	3.16e-01	-1.00

Table 7: Dunn-Bonferroni test for Subgroups between Precincts

Figure 2 is used to visualize the distribution of the aggregated datasets using boxplots. The median observation time at Precinct 1 is shifted slightly to the left compared to the other precincts. It is noted that practically all the observations are within the interquartile range of all other precincts.

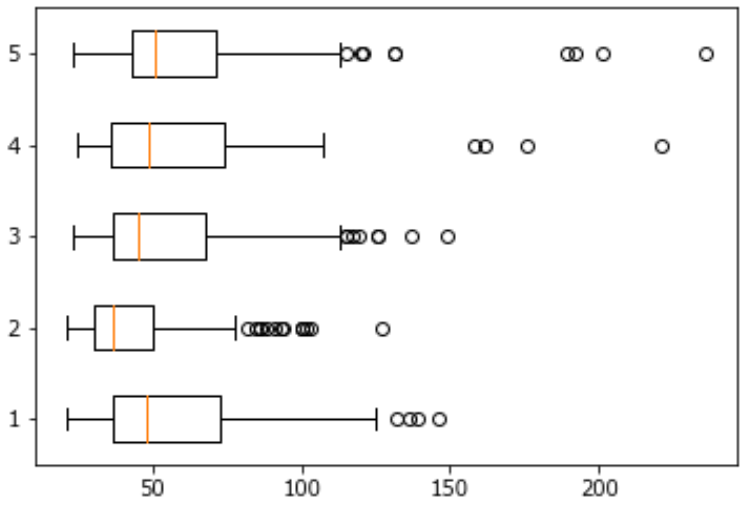

Figure 2: Boxplot of Check-in Observation Times by Precinct

4.2 POISSON HIDDEN MARKOV MODELS

Arrival rates for four precincts in Providence Rhode Island at 15-minute intervals using a dt-PHMM. After the prescreening procedure is completed, a stochastic process model is created using observations from the 2018 Midterm time study. A dt-PHMM then constructed for each precinct using three, four, and five hidden states. The fitness of each model and output parameters are discussed.

4.2.1 PRECINCT 1

Approximately 38% of the 3222 registered voters casted their ballot at Precinct 1 Election day. Processing times at the four check-in stations were observed between 10:30AM-12:30 PM and averaged 53 seconds. The Log KS test returned a p-value of 0.027 for arrivals between 10:45 AM-11:00 AM. However, the null hypothesis failed to be rejected after the FDR correction procedure was applied. Comparing the arrival counts to the 75% throughput capacity in Figure 3 shows that enough capacity existed to prevent significant queues from forming before the check-in station.

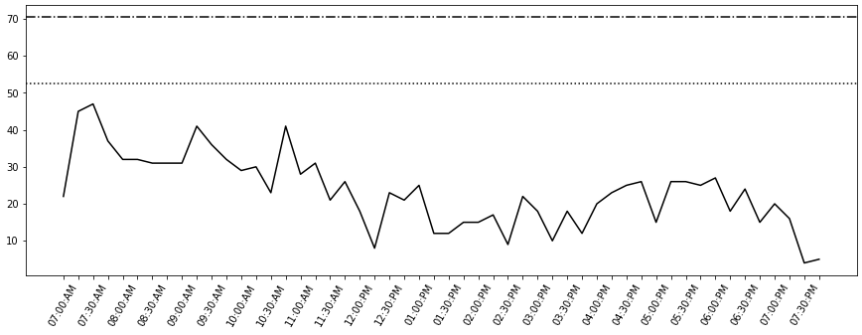

Figure 3: Arrival Rate vs. Check-in Capacity for Precinct 1

Initial estimates for λ_0 are generated for each model by applying the k-means clustering algorithm to the static arrival sequence. Twenty observation sequences generated from the EPB timestamps using a stochastic processing times generated from a log normal distrution based on the parameter values given in Table 2. Table 6 gives the initial and fitted estimates for rate parameters λ_0 , $\hat{\lambda}$, and $\hat{\boldsymbol{\theta}}$, the transition probability matrix.

STATE MODEL	**M = 3**	**M = 4**	**M = 5**
$\boldsymbol{\lambda_0}$	[41.2 26.1 13.2]	[41.2 28.3 18.9 9.0]	[43.5 31.7 24.1 17.1 9.0]
$\hat{\boldsymbol{\lambda}}$	[34.1 22.9 13.4]	[42.4 31.4 22.9 13.3]	[42.8 31.5 23.8 16.4 4.51]
$\hat{\boldsymbol{\theta}}$	$\begin{bmatrix} 0.983 & 0.164 & 0 \\ 0.052 & 0.839 & 0.109 \\ 0 & 0.103 & 0.897 \end{bmatrix}$	$\begin{bmatrix} 0.580 & 0.420 & 0 & 0 \\ 0.107 & 0.879 & 0.014 & 0 \\ 0 & 0 & 0.882 & 0.118 \\ 0 & 0 & 0.109 & 0.891 \end{bmatrix}$	$\begin{bmatrix} 0.592 & 0.408 & 0 & 0 & 0 \\ 0.103 & 0.879 & 0.018 & 0 & 0 \\ 0 & 0 & 0.937 & 0.063 & 0 \\ 0 & 0 & 0.019 & 0.932 & 0.049 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$
LOG LIKELIHOOD	-3543.37	-3495.25	-3407.92
BIC	-3560.74	-3519.57	-3439.18
AIC	-3548.38	-3502.25	-3416.92

Table 8: HMM Parameter Estimates for Precinct 1

The fitted $\hat{\lambda}$ values corresponding to Viterbi state sequence plotted against arrival counts for the first and second models (m=3, m=4) in Figure 4 to draw qualitative comparisons between the models.

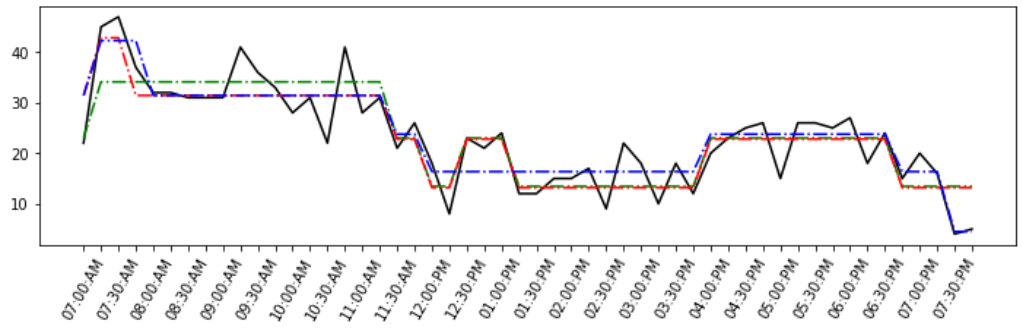

Figure 4: Viterbi Sequence vs. Observation Sequence for Precinct 1

The third model (m=5) is not considered to be valid due to the transition probability for the fifth state converging to 1. The 4-state model is marginally better than the 3-state in terms of the BIC and AIC scores. The additional state-space allows the 4-state model to account for the peak arrival period in the early morning. Nevertheless, the Viterbi state sequences for both models appear to be an accurate representation of the observation sequences.

4.2.2 PRECINCT 2

Precinct 2 has 3218 registered voters and experienced ~53% on Election day. Check-in processing time observations were collected at this precinct between 11:30AM-2:55 PM. The four check-in servers had an average processing time of 42 seconds.

A static series of arrival times are generated by subtracting the average processing time from each EPB timestamp. Applying the Log KS test to the data in 15-minute intervals initially rejects the intervals with P-values less than 0.05 listed in Table 7. This rejection is overturned using the FDR correction procedure. It is concluded that pseudo-start times adequately represent a Poisson process.

Time Interval	P-value
7:15-7:30 AM	0.0153
7:45-8:00 AM	0.0426
8:15-8:30 AM	0.0153
12:15-12:30 PM	0.0451
3:30-4:00 PM	0.0344
4:15-4:30 PM	0.0299
6:45-7:00 PM	0.0294

Table 9: P-values less than 0.05 for Log-KS test for Precinct 2

The average observed processing time is used to calculate a deterministic throughput rate, shown by the dashed line in Figure 3. The dashed line displays 75% of the deterministic throughput capacity. Inspection of Figure 5, showing the arrival count vs. estimated capacity, provides evidence to support the assumption that start times may be used as an arrival time proxy because the wait time to check-in is negligible.

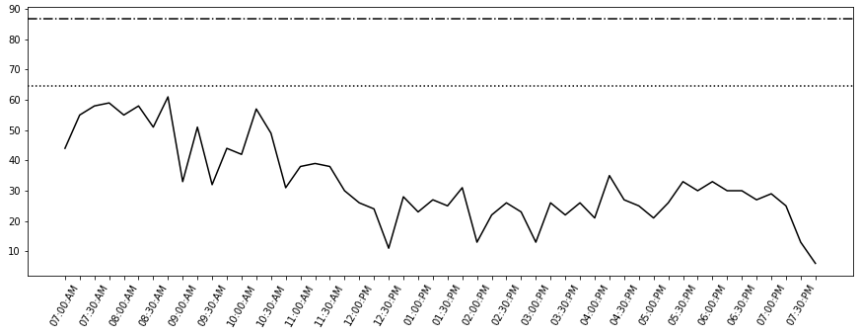

Figure 5: Arrival Rate vs. Check-in Capacity for Precinct 2

The static arrival sequence is clustered using the k-means algorithm. Cluster centers are used as initial estimates for the arrival rate parameter λ_0. A stochastic processing time model is used to generate an arrival observation sequence $\{O_t\}$. A total of 20 samples sequences are used to train each model. Table 8 gives the initial and fitted parameter estimates for rate parameters λ_0 and $\hat{\lambda}$ as well as the transition probability matrix.

STATE MODEL	**M = 3**	**M = 4**	**M = 5**
$\boldsymbol{\lambda_0}$	[52.6 28.3 11.2]	[55.4 37.4 26.2 11.2]	[55.4 40.8 31.2 24.6 11.2]
$\hat{\boldsymbol{\lambda}}$	[50.3 28.5 11.1]	[56.6 42.3 27.4 11.2]	[56.6 44.2 36.6 26.7 11.2]
$\hat{\boldsymbol{\theta}}$	$\begin{bmatrix} 0.925 & 0.075 & 0 \\ 0.030 & 0.850 & 0.120 \\ 0 & 0.740 & 0.260 \end{bmatrix}$	$\begin{bmatrix} 0.825 & 0.175 & 0 & 0 \\ 0.130 & 0.843 & 0.026 & 0 \\ 0 & 0.03 & 0.839 & 0.128 \\ 0 & 0 & 0.739 & 0.261 \end{bmatrix}$	$\begin{bmatrix} 0.891 & 0.109 & 0 & 0 & 0 \\ 0.115 & 0.833 & 0.052 & 0 & 0 \\ 0 & 0 & 0.868 & 0.132 & 0 \\ 0 & 0 & 0 & 0.868 & 0.132 \\ 0 & 0 & 0 & 0.734 & 0.266 \end{bmatrix}$
LOG LIKELIHOOD	-3773.50	-3728.91	-3505.45
BIC	-3790.86	-3753.22	-3536.71
AIC	-3778.49	-3735.91	-3514.45

Table 10: HMM Parameter Estimates for Precinct 2

The fitted $\hat{\lambda}$ values corresponding to Viterbi state sequence plotted against arrival counts for each model in Figure 6 to draw qualitative comparisons between the models.

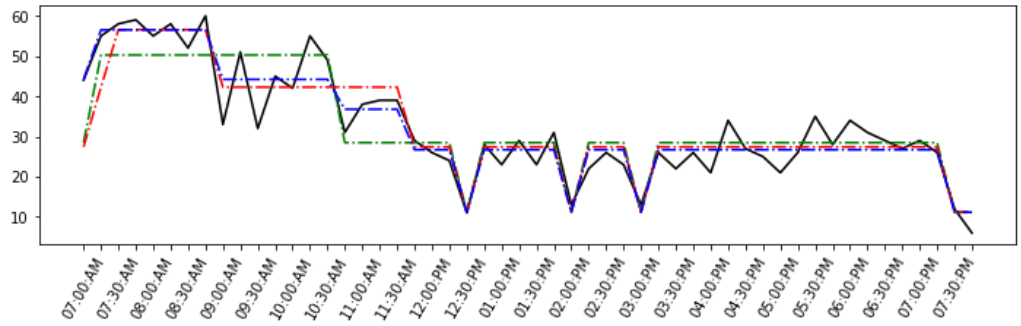

Figure 6: Viterbi Sequence vs. Observation Sequence for Precinct 2

The BIC and AIC values indicate the dt-PHMM with five states (m=5) has the highest probability of observing the training sequences. The Viterbi state sequences for all three models is nearly identical for all three models. Excluding the low points at 12:15, 1:45, and 3:00, the observation sequence appears relatively stable during the second half of the day. The first and second models (m=3, m=4) lack the state space to account for the local variability compared to the 5-state model.

4.2.3 PRECINCT 3

Approximately 55% of this precinct's 3130 registered voters casted their ballot at Precinct 3 Election day. Check-in processing times from all four stations were collected from 6:30 PM-7:30 PM. The average observation time of 51 seconds was subtracted from each EPB timestamp to generate a static series of arrival times. The 7:45 PM-8:00 PM time block was the only time interval rejected the null hypothesis for the Log- KS test with a P-value = 0.0176 but was deemed to be a false positive after FDR correction.

The deterministic throughput rate and 75% throughput capacity estimates were calculated based on a check-in process time of 51 seconds. The static arrival counts plotted in Figure 7 exceed the 75% capacity several times but do reach 100%. The negligible wait assumption is upheld by poll worker testimony that line formation was not significant at any point during the day.

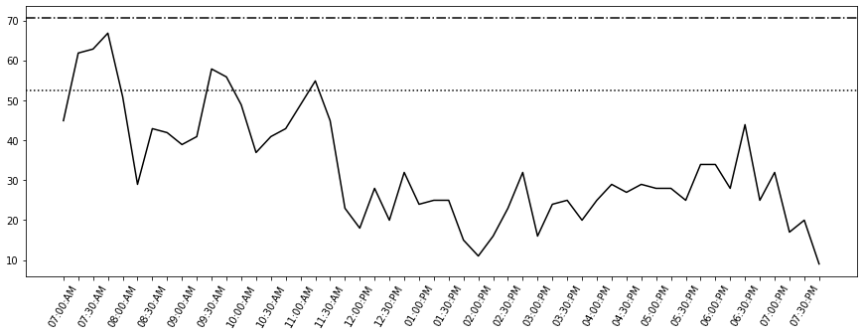

Figure 7: Arrival Rates vs. Check-in Capacity for Precinct 3

Initial estimates for the rate parameter λ_0 were computed based on the k-means clustering algorithm for dt-PHMM models with three, four, and five states. A total of 20 observation sequences were generated from the EPB timestamps based on a stochastic process time model. Table 9 gives the initial and fitted parameter estimates for rate parameters λ_0 and $\hat{\lambda}$ as well as the transition probability matrix.

STATE MODEL	**M = 3**	**M = 4**	**M = 5**
$\boldsymbol{\lambda_0}$	[56.7 38.9 22.6]	[60.2 43.9 27.5 16.3]	[61.2 44.9 34.4 25.8 15.8]
$\hat{\boldsymbol{\lambda}}$	[57.4 39.7 23.3]	[60.5 45.7 27.4 15.0]	[61.8 46.9 40.7 26.7 14.8]
$\hat{\boldsymbol{\theta}}$	$\begin{bmatrix} 0.694 & 0.306 & 0 \\ 0.155 & 0.720 & 0.120 \\ 0 & 0.008 & 0.992 \end{bmatrix}$	$\begin{bmatrix} 0.511 & 0.489 & 0 & 0 \\ 0.118 & 0.815 & 0.067 & 0 \\ 0 & 0.043 & 0.899 & 0.058 \\ 0 & 0 & 0.144 & 0.856 \end{bmatrix}$	$\begin{bmatrix} 0.776 & 0 & 0.224 & 0 & 0 \\ 0 & 0.918 & 0 & 0.082 & 0 \\ 0.237 & 0.118 & 0.645 & 0 & 0 \\ 0 & 0 & 0 & 0.947 & 0.053 \\ 0 & 0 & 0 & 0.136 & 0.864 \end{bmatrix}$
LOG LIKELIHOOD	-3773.97	-3851.27	-3584.45
BIC	-3791.34	-3875.59	-3615.82
AIC	-3778.97	-3858.27	-3593.55

Table 11: HMM Parameter Estimates for Precinct 3

The fitted $\hat{\lambda}$ values corresponding to Viterbi state sequence plotted against arrival counts for each model in Figure 8 to draw qualitative comparisons between the models.

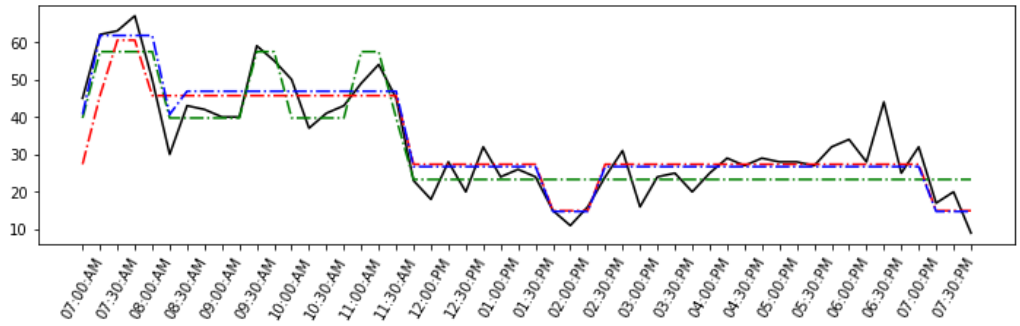

Figure 8: Viterbi Sequence vs. Observation Sequence for Precinct 3

The first model (m=3) fits the observed counts comparatively well for the first half of the sequence but fails to account for any variation or trend in the second half of the day. The second model (m=4) underestimates the early morning peak but appears to be a good fit for the rest of the day. The BIC and AIC scores indicate that the 5-state model performs best. However, the third state is only used twice in the Viterbi sequence. Furthermore, the fitted value $\hat{\lambda}_2 = 46.9$ is very close to fitted value $\hat{\lambda}_3 = 40.7$ suggesting that using five states may be superfluous under the current model.

4.2.4 PRECINCT 4

Approximately 24% of the 3276 registered voters in Precinct 4 casted their ballot in-person Election day in 2018. Check-in processing time observations were collected for the five stations between 7:00 AM-9:00 AM. period. One of the check-in stations was only utilized three times during the observation period. These observations were not included in the analysis because two of them were instances where the voters casted a provisional ballot which required additional services from the clerk. The other three stations had an average observed time of 55 seconds. After

subtracting the average processing time from the EPB timestamps, the Log KS found that intervals listed in Table 10 had P-values less than 0.05. Application of the FDR correction procedure concluded theses intervals to be false positives. The null hypothesis that emissions from all intervals are a Poisson process failed to be rejected for the entire dataset.

Time Interval	P-value
1:15-1:30 PM	0.0385
1:30-1:45 PM	0.0176
6:00- 6:15 PM	0.0017
6:15-6:30 PM	0.0432

Table 12: P-values less than 0.05 for Log-KS test for Precinct 4

The arrival counts for each 15-minute interval are well below the deterministic throughput rate, and 75% capacity displayed in Figure 9.

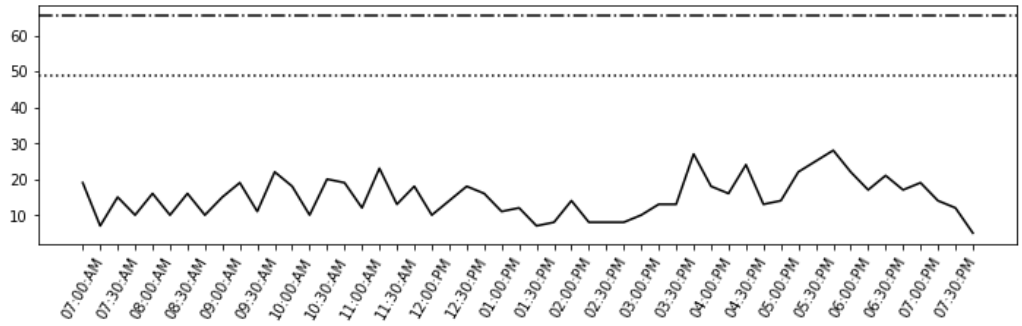

Figure 9: Arrival Rate vs. Check-in Capacity for Precinct 4

The initial rate parameter values, λ_0 , are shown in Table 11 for all three models. A total of 20 observation sequences were generated to train the models. The fitted values for the arrival rate and transition probability matrixes are provided in Table 11 for the three-state model. The Baum-Welch algorithm failed to converge to exact parameter values for $\hat{\theta}$ and in the four and five-state models. Therefore, the values listed in Table 11 are mere approximations.

STATE MODEL	M = 3	M = 4	M = 5
$\boldsymbol{\lambda_0}$	[23.4 16.6 10.0]	[23.8 17.7 13.1 8.5]	[23.8 17.7 13.1 8.5 8.3]
$\hat{\boldsymbol{\lambda}}$	[20.1 15.1 9.6]	[25.4 20.1 15.2 9.6]	[24.1 16.9 11.6 4.3 4.2]
$\hat{\boldsymbol{\theta}}$	$\begin{bmatrix} 0.988 & 0.012 & 0 \\ 0.002 & 0.986 & 0.012 \\ 0 & 0.022 & 0.978 \end{bmatrix}$	$\begin{bmatrix} 0.626 & 0.374 & 0 & 0 \\ 0 & 0.989 & 0.011 & 0 \\ 0 & 0.002 & 0.986 & 0.012 \\ 0 & 0 & 0.019 & 0.981 \end{bmatrix}$	$\begin{bmatrix} 0.871 & 0.129 & 0 & 0 & 0 \\ 0.008 & 0.981 & 0.011 & 0 & 0 \\ 0 & 0.017 & 0.983 & 0 & 0 \\ 0 & 0 & 0 & 0.798 & 0.202 \\ 0 & 0 & 0 & 0.752 & 0.248 \end{bmatrix}$
LOG LIKELIHOOD	-3238.02	-3269.28	-3247.02
BIC	-3255.39	N/A	N/A
AIC	-3243.02	N/A	N/A

Table 13: HMM Parameter Estimates for Precinct 4

The arrival counts are plotted against the arrival rates corresponding to the Viterbi state sequence for the 3-state model in Figure 10.

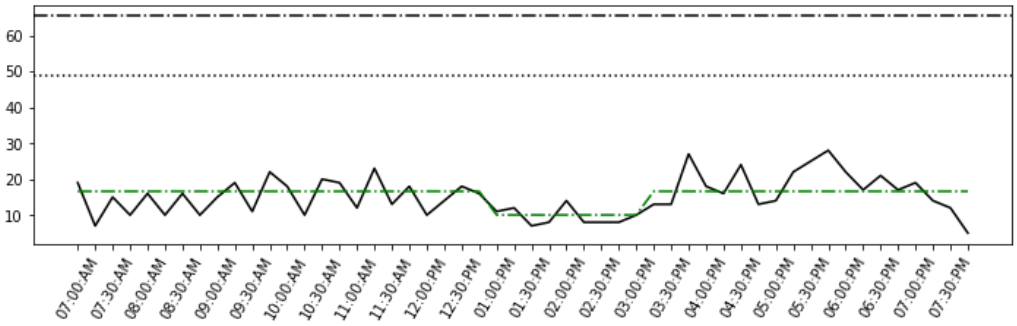

Figure 10: Viterbi Sequence vs. Observation Sequence for Precinct 4

After inspecting the most likely state sequence provided by the Viterbi algorithm for the 3-state model, it is noted that only the two states corresponding with lower rates are predicted in the Viterbi state sequence. The state with the highest arrival rate is not reached despite the apparent increase in arrivals in the latter part of the day.

CHAPTER 5

CONCLUSION

5.1 CHECK-IN PROCESSING TIME CONCLUSIONS

The comparative analysis within precincts demonstrated similar results for most precincts. The check-in station observations at individual precincts were statistically similar to one another at five out of seven locations. At Precinct 5, three out of the four stations were statistically similar while the fourth averaged ~10 seconds faster. Precinct 6 check-in stations split into two subgroups of similar observations. Additional information about the polling place layout and volunteer testimonials are needed to speculate the underlying factors causing these differences. Nevertheless, using a single process model to represent all check-in stations is concluded to be a reasonable assumption.

The comparative analysis between precincts concluded that the check-in processing time at 4 out of 5 locations are statistically similar. Observations at the fifth location were only five seconds faster on average. It is tentatively concluded that the use of a generalized check-in process model would not be an unreasonable assumption when constructing Hidden Markov Models from EPB data where precinct specific observations are not available.

5.2 HIDDEN MARKOV MODEL CONCLUSIONS

A discrete-time Hidden Markov model was successfully able to estimate arrival rates of four precincts. The most appropriate size for the hidden state-space varied between precincts. The strengths and weakness of the three, four, and five-state models are discussed for each precinct.

At Precinct 1, the three and four-state models were valid. The Viterbi sequence in both cases produced similar results models. The five-state model was rejected due to the transition probability in the last state fully converging to a single value. Arrivals at Precinct 1 dropped off significantly after 7:15 PM. It is speculated that the additional state-space in this model was used to account for these extremely low values at the end of the sequences. The inclusion of a lower bounding constraint of some sort would be beneficial for this specific model.

The dt-PHMMs constructed for Precinct 2 are considered valid models for the three, four, and five-state cases. The Viterbi state sequences for three and four-state models estimated the observation sequences reasonably well. The five-state performs considerably better than the previous two models in term of their AIC and BIC scores. The additional state-space allows the five-state model to better account for the variability throughout the day-especially in the late morning period.

The dt-PHMMs constructed with three and four hidden states are valid for Precinct 3. The Viterbi sequence produced by the three-state model was better able to represent the three peaks during the first half of the day but grossly underestimated the variability later. The Viterbi sequence of the four-state model performed moderately well in the morning but did a far better just depicting the variability in the second half of the day. The additional state-space in the last model is considered to superfluous because the third state is only seen twice in the Viterbi sequence. Furthermore, there is no meaningful difference between $\hat{\lambda}_2$ and $\hat{\lambda}_3$.

Arrivals to Precinct 4 were unique in comparison to the other precincts. The three-state dt-PHMM was the only model that fully converged in this test case. It is

noted that the state with the highest arrival rate is not reached in the Viterbi sequence despite an apparent increase in arrivals later in the day. The apparent oscillation in the observation sequence suggests that a continuous-time model would be more appropriate in this case. It is concluded that the observation sequence is marginally aperiodic, and the discrete-time model is still considered to be a valid model.

5.3 LIMITATIONS

There are, however, some limitations inherent to the modeling assumptions used in this research. Firstly, the procedure used to ascertain arrival times is only valid when the arrival rates are less than the overall throughput capacity of the voting system. The processing time distributions are based solely on observations of the standard check-in process and do not include times where voters required additional services. Although these cases occur infrequently, they can last significantly longer than the standard check-in process. The 75% capacity is used in the second prescreening to account for the possibility of one of the check-in stations being occupied by one voter for the entire 15-minute segment. There is also an implicit assumption that the check-in station is the bottleneck of the operation. While this assumption has historically been accepted, the implementation of EPB has dramatically changed the way voters flow throughout the system (Haas, 2014). A more robust procedure for validating these assumptions is needed in order to increase the extensibility of this work.

The second limitation stems from the requirement that emissions of Poisson processes must strongly stationary and exponentially distributed. The FDR correction (Benjamini & Hochberg, 1995) is a less conservative procedure, thereby relaxing the

overdispersion constraint. While this is acceptable while using pure count data (Paroli, Redaelli, & Spezia, 2002) there is still some risk in the assumption that state-changes occur at discrete intervals.

The third limitation is that a theoretically sound procedure for model validation has yet to be established (Paroli, Redaelli, & Spezia, 2002). A residual analysis cannot be performed because it is not possible to compute the residuals from an unobserved Markov chain (Paroli, Redaelli, & Spezia, 2002). Additional time studies (collecting actual arrivals over the entire day) are required to test this work within a broader scope of use-cases.

5.4 FUTURE WORK

This research study establishes a baseline procedure for estimating voter arrival behaviors through Hidden Markov Models. The scope of future work on the immediate horizon will focus on reducing the workload to instantiate and train new model instances. A bootstrapping method will be explored for automatically adjusting input parameter values when the Baum-Welch algorithm fails to converge. Alternative implementations including but not limited to continuous-time hidden Markov models, time-dependent hidden Markov models, mixture models, and Kalman Filters.

Future work in the near term will also focus on creating a more robust prescreening procedure. The data from individual Poll Pads will be used to explore the relationship between the mean and variance of time between successive observations in each time block. When the arrival rate is well below throughput capacity the time in between successive timestamps are expected to be exponentially distributed. As the arrival rate increases, the mean and variance will decrease proportionately until a

queue begins to form. As the arrival rate reaches throughput capacity, a new voter will likely arrive almost immediately after a poll pad becomes available. In this case, the time deltas are expected to come from the same lognormal distribution as the check-in process time. A Ratio of Maximized Likelihood (RML) test illustrated in Gupta, Rameshwar, & Kundu (2005) can be used to discriminate between a Lognormal or Generalized Exponential distribution. This will be useful when developing experimental missing data methods for estimating the arrival rates beyond the threshold of the check-in capacity station.

APPENDICES

APPENDIX A

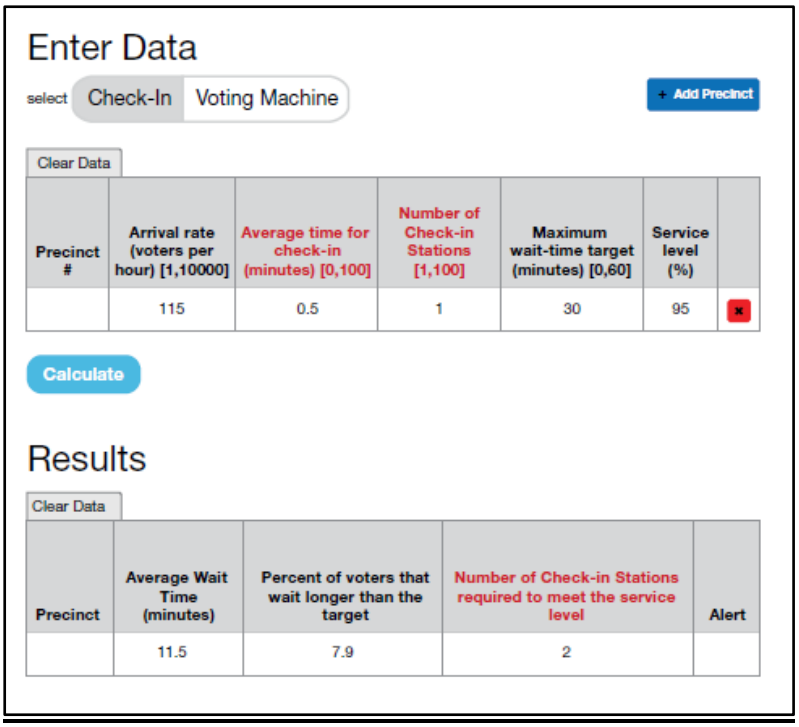

Figure A1: Voter Technology Project: Graves-Yuan Tool

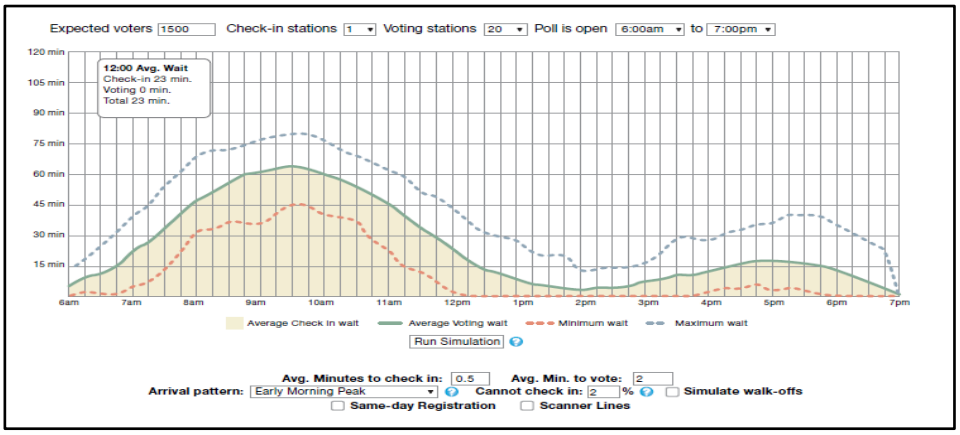

Figure A2: Voter Technology Project: Pelczarski Tool

APPENDIX B

Data Collection Instructions (2018)

General Notes

- Meet with the Moderator and ask where you can stand/sit for data collection (try to make sure you have a view to all stations).

- Locate the station you plan to observe (i.e., Check-in poll pads, voting booths, scanners).
- DO NOT GO PAST THE CHECK-IN AREA!
- Try to track as many voters as you can throughout the three hours. If you miss a voter entering your station, skip that observation. If you lose track of the voter, use the Undo Last feature to remove the observation.

Data Collection by Station

- Arrival and Random Sampling
    - As voters arrive at the polling place (enter the room in which voting takes place), click the Voter Arrival button on the calculator.
    - As frequently as possible, track a voter throughout the entire voting system. Do this by clicking start as they arrive (as defined above). When the voter finishes scanning the ballot, click the Stop button for that voter.
    - There are two timers to track voters throughout the voting system. A text box is provided to input identifiers so that voters are tracked consistently.
- Check-in
    - Once the voter is called up by the supervisor or approaches the check-in table, click the Start button on the timer.
    - Once the voter has received their ballot, is sent away, or moves to the clerk, click the Stop button.
    - Keep track of each poll pad consistently, so that poll pad one on the spreadsheet always has observations from the same poll pad in use. If there are several poll pads in the polling place, number them from left to right before data collection and use this consistently throughout.
- Voting booth
    - As soon as a voter approaches a booth, click the Start button on the timer.
    - When the voter exits the booth (when they begin to walk away), click the Stop button.
    - There will likely be many voting booths at the polling location. This timer allows you to track up to five voters at a time.
    - Use the text box field to input identifiers to help you keep track of which voter is which.
    - The number on the timer does not need to be assigned to specific booths (like check-in) but rather to a specific voter.
- Scanner
    - As soon as the voter approaches the DS200 scanning machine, click the Start button on the timer.
    - When the voter begins to walk away from the scanner, click the Stop button.

    - If a voter has an error and must correct the ballot (walks away but does not exit the polling location) press stop. Their next scanning attempt will be treated as a new observation.
    - If anything unusual happens (machine breakdowns, technicians fixing scanner, etc.) try to take a note of this.
- Completing Data Collection
    - At the end of the three hours, save the Excel file with your name, station, and the precinct at which you collected data ("LastnameStationPrecinctNumber.xlsm").

If anyone needs to use the restroom, have another member track your station in your absence

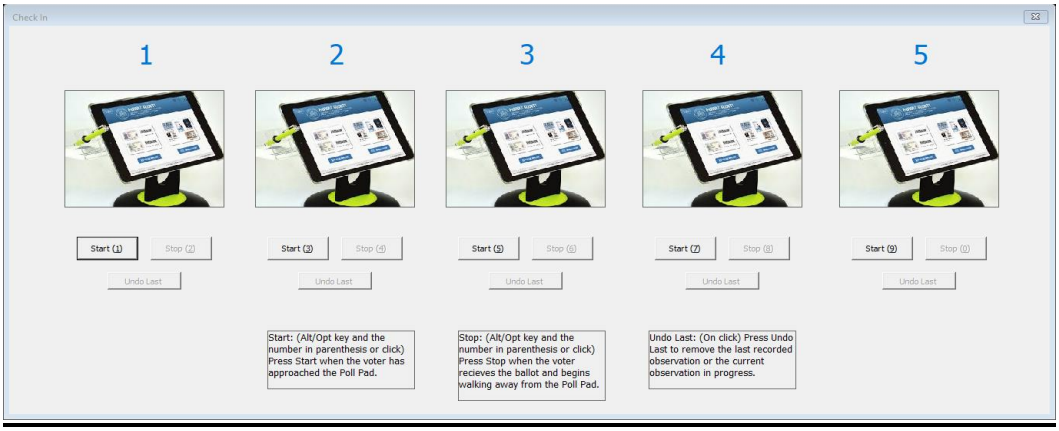

Figure A.3: Check-In process timer used in the 2018 Time Studies

BIBLIOGRAPHY

Allen, T., & Bernshteyn, M. (2006). Mitigating Voter Wait Times. *Chance Magazine*, pp. 25-36.

Baum, L., Petrie, T., & Weiss, N. (1970). A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov chains,. *The Annals of Mathematical Statistics*, pp. 164-171.

Benjamini, Y., & Hochberg, Y. (1995). Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society*, 289-300.

Brown, L., Gans, N., Mandelbaum, A., Sakov, A., Shen, H., Zeltyn, S., & Zhao, L. (2006). Statistical Analysis of a Telephone Call Center.

Dow, K. (2007). *Study of VOter Flow at the 2006 General Election, Columbia NY.*

Edelstein, W. (2006). *New Voting Systems for New York- Long Lines and High Cost.*

Edelstein, W. A., & Edelstein, A. D. (2010). Queuing and Elections: Long Lines, DREs and Paper Ballots. *Electronic Voting Technology Workshop/Workshop on Trustworthy Elections.* Washington, DC.

Feldman, D., & Belcher, C. (2005). *Voting Experience Survey.* Democratic National Committee.

Fortier, J. C., Stewart III, C., Pettigrew, S., Weil, M., & Harper, T. (2018). *Improving the Voter Experience; Reducing Polling Place Wait Times by Measuring.* Bipartisan Policy Center.

Gupta, Rameshwar, W., & Kundu, D. (2005). Discriminating between Lognormal and Generalized Exponential Distributions. *Journal of Statistics & Data Analysis*, 213-227.

Haas, M. (2014). *Electronic Poll Book Research – Final Report.* Madison, Wisconsin.

Herron, C. M., & Smith, D. A. (2015). Precinct Resources and Voter Wait Times. *New Research on Election Administration and.* Cambridge, MA.

Kruskal & Wallis. (1952). Use of Ranks in One-Criterion Variance Analysis. *Journal of the American Statistical Association*, 583-621.

Leroux, M., & Puterman, M. (1992). Maximum-Penalized-Likelihood Estimation for Independent and Markov-Dependent Mixture Models. *Biometrics*, 545-558.

Paroli, R., Redaelli, G., & Spezia, L. (2002). POISSON HIDDEN MARKOV MODELS FOR TIME SERIES. 461-474.

Rabiner. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *IEEE*, 257-286.

Rabiner, L., & Juang, B. (1986). An Introduction to Hidden Markov Models. *IEEE ASSP Magazine*, pp. 4-16.

Scott. (2001). Detecting Network Intrusion Using the Markov Modulated Nonhomogeneous Poisson Process. *Journal of the American Statistical Association*.

Spencer, D. M., & Markovits, Z. S. (2010). Long Lines at Polling Stations? Observations from an Election Day Field Study.

Stewart III, C. (2015). *Managing Polling Place Resources.* Caltech/MIT Voting Technology Project.

Tiberiu, C., & Harrison, P. (2013). Analyzing and Predicting Patient Arrival Times in Hospitals using Hidden Markov Models. *Information Sciences and Systems,* (pp. 77-85).

Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 260-269.

Yang, M., Fry, M. J., & Kelton, D. W. (2009). Are All Voting Queues Created Equal? *Winter Simulation Conference.* IEEE.

Yang, M., Kelton, D., Fry, M. J., & Allen, T. T. (2013). Improving Voting Systems through Service-Operations Management. *Production and Operations Management*.