MACHINE LEARNING-BASED HIGH FIDELITY MESOSCOPIC MODELING TOOL FOR TRAFFIC NETWORK OPTIMIZATION

FINAL PROJECT REPORT

by

Robert B. Heckendorn, Neeta A. Eapen, and Ahmed Abdel-Rahim
University of Idaho

Sponsorship
PacTrans and the University of Idaho

for
Pacific Northwest Transportation Consortium (PacTrans)
USDOT University Transportation Center for Federal Region 10
University of Washington
More Hall 112, Box 352700
Seattle, WA 98195-2700

In cooperation with US Department of Transportation-Research and Innovative Technology Administration (RITA)

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The Pacific Northwest Transportation Consortium, the U.S. Government and matching sponsor assume no liability for the contents or use thereof.

Technical Report Documentation Page

1. Report No.	2. Government Accession No. 01784881	3. Recipient's Catalog No.
4. Title and Subtitle Machine Learning-Based High Fidelity Mesoscopic Modeling Tool for Traffic Network Optimization		5. Report Date 6. Performing Organization Code
7. Author(s) Robert B. Heckendorn, 0000-0002-1319-0670; Neeta A. Eapen, and Ahmed Abdel-Rahim, 0000-0001-9756-554X		8. Performing Organization Report No. 2021-S-UI-3
9. Performing Organization Name and Address PacTrans Pacific Northwest Transportation Consortium University Transportation Center for Region 10 University of Washington More Hall 112 Seattle, WA 98195-2700		10. Work Unit No. (TRAIS) 11. Contract or Grant No. 69A355174110
12. Sponsoring Organization Name and Address United States of America Department of Transportation Research and Innovative Technology Administration		13. Type of Report and Period Covered 14. Sponsoring Agency Code
15. Supplementary Notes Report uploaded at www.pacTrans.org		
16. Abstract Traffic signal control of the future will adapt in real time to traffic, driving, and environmental conditions. The adaptations will require real-time optimization based on traffic monitoring. However, for optimization to work, it must be able to predict what traffic will do in response to changes in signal timing. Furthermore, it must do this prediction very efficiently, in a short time horizon, so that the optimization process can evaluate many alternative signal timing parameters. Instead of using a "one size fits all" legacy function for prediction of traffic flow, this project took steps to pioneer the use of machine learning techniques to learn how traffic behaves on particular segments of streets in response to traffic and driving conditions and used that learning to build a high-speed simulator to predict the arrival times of cars at intersections.		
17. Key Words Traffic simulation, traffic optimization, machine learning		18. Distribution Statement No restrictions.

19. Security Classification (of this report)	20. Security Classification (of this page)	21. No. of Pages	22. Price
Unclassified.	Unclassified.		NA

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

CONTENTS

Figures

Tables

EXECUTIVE SUMMARY

A clear application of computing to traffic is real-time optimization of traffic signal operations. However, this kind of problem requires an evaluation of the quality of various potential solutions. While this is not difficult, it is currently extremely slow, rendering real-time solution infeasible.

In this project, we proposed replacing current micro-simulation solutions with a meso-simulator that uses machine learning to learn the arrival time functions for traffic parameters on a street segment with street segment characteristics. We ran a priority queue-based simulator that estimated the arrival time of each car at the next decision point for that car. We tested our simulator against data collected from the widely used VISSIM micro-simulator. A big advantage of this solution is the customization of the arrival time predictor for specific street segments and other parameters such as time of day or weather. This would replace the one-size-fits-all formulas used by most simulators.

This page is intentionally left blank.

1 INTRODUCTION AND APPROACH

1.1. Background and Introduction

Traffic optimization is a complex system without reliable and situation-specific closed form formulas for optimization. This means that simulation plays a critical role in any advancement in traffic optimization. Micro-models simulate car behavior in very small steps, leading to extremely long computing times that are impractical for any in-depth optimization. Furthermore, the connection with real traffic behavior is tenuous because the car behavior model is based on a few simple, hard coded abstractions that apply everywhere. This affects the model's reliability to predict traffic flow. In contrast, meso-simulation is much faster because it requires only the ability to estimate arrival times at the ends of street segments. Unfortunately, these estimations are difficult to make because they must be based not only on engineering parameters but also on road conditions, including congestion and subjective influences such as width and weather conditions.

With the advent of specialized hardware and high-performance bus and memory architectures, machine learning has shown near magical improvement in the last 10 years. Evidence of this is plainly visible on the cell phones we use every day, such as speech recognition and classification of a picture library by subject. Machine learning has been applied to various traffic problems as well, including methods of predicting congestion [3, 5, 6] using techniques such as random forests and neural networks. Other examples are intersection traffic prediction [2], traffic density prediction using long short-term memory (LSTM) networks [1], traffic forecasting with Bayesian networks [8], and accident prediction accomplished by distinguishing normal from abnormal traffic behaviors [4]. We applied this machine learning technology to predicting the behavior of traffic on individual road segments, thereby allowing us to quickly assess arrival times.

1.2. Research Approach

We produced a fast mesoscopic simulation whose key component was a predictor function for each road segment to predict the travel time distribution for each car across that segment. Let $d_s(c, \theta)$ be a predictor function for a segment *s,* where *c* is the congestion on the segment and θ is a vector of other factors such as road conditions and weather conditions. Let *p* denote the travel time distribution predicted by the predictor function. $d_s(c, \theta)$ predicts the travel time distribution *p*. During the running of the simulation, random samples from the distribution *p* are used.

To obtain these predictor functions, we proposed using machine learning techniques based on monitoring of real traffic behavior. Specifically, we aimed to test our ideas by using machine learning to learn predictor functions for the streets in various model cities run with the popular VISSIM micro-simulator. That is, VISSIM would stand in for real-world traffic observations. To measure our success, we would compare the timings seen in VISSIM with those in our mesoscopic simulator. Furthermore, because the mesoscopic simulator would not have to simulate every small step along a road, it would run much faster. Perturbation of VISSIM traffic data controlled by auxiliary parameters would allow us to test more difficult traffic simulation conditions.

An advantage of our approach was that functions could be street- and time-specific. This could include even behaviors that standard simulators might find difficult to handle, such as optically narrow streets, 5:00 pm traffic that faces a setting sun, intersections with complex characteristics based on awkwardly sized turn lanes, etc. This specificity was made possible by monitoring actual traffic behavior and using machine learning methods to build a predictor for that case. The overall result would be simulations that run at greatly increased speed and fidelity.

2. WORK PERFORMED AND RESULTS

2.1. Simulation Model Overview

Using the VISSIM micro-simulator as a benchmark for "reality," we built a mesoscopic-level simulator with the objective of comparing its performance and accuracy. The basic simulator was constructed but required learning from collected VISSIM data, which we did. Our first machine learning work was to create a neural network function that used street segment parameter data to generate an arrival time as an output. Unfortunately, this was not what we actually observed in the real world or as an output of VISSIM. What we actually got was a distribution of arrival times. An example of this can be seen in the simple function shown in Figure 2-1 mapping congestion to travel time taken from observations of travel time on a street segment in VISSIM.

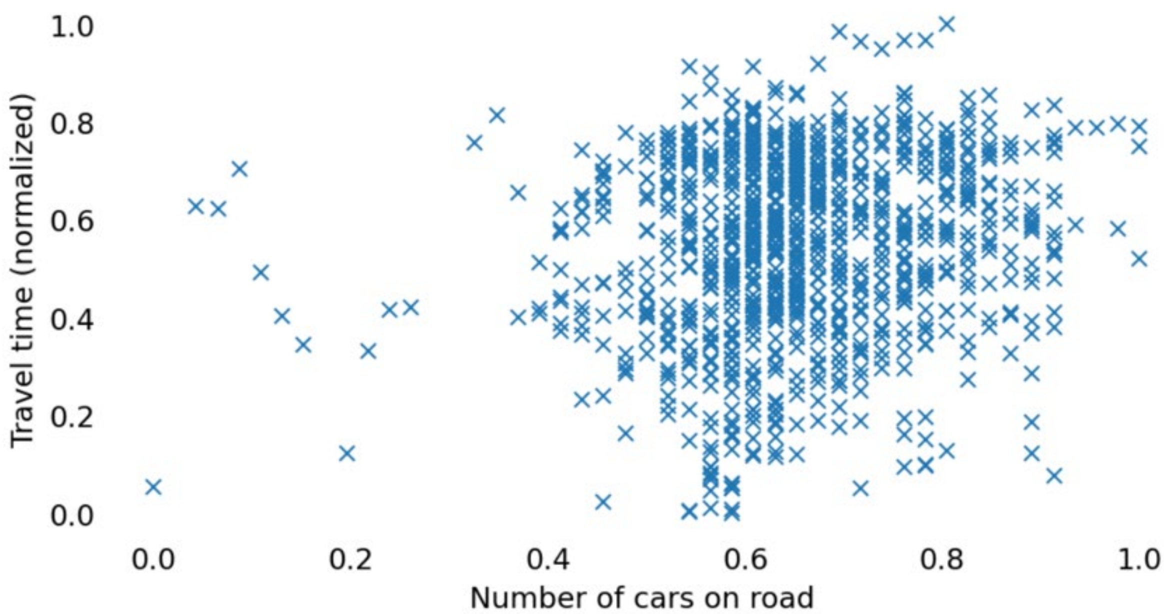

Figure 2-1 Normalized travel time versus number of cars on the road

We tried and discarded the idea of learning a normally distributed function of arrival times. Instead, we developed a Gaussian Mixture Model approach for the non-Gaussian distributions we were observing. In the example of a skewed arrival time distribution shown in Figure 2-2, our algorithm created a mixture model that fit the target distribution and allowed us to create a random distribution function that was quick to evaluate for use in the simulator.

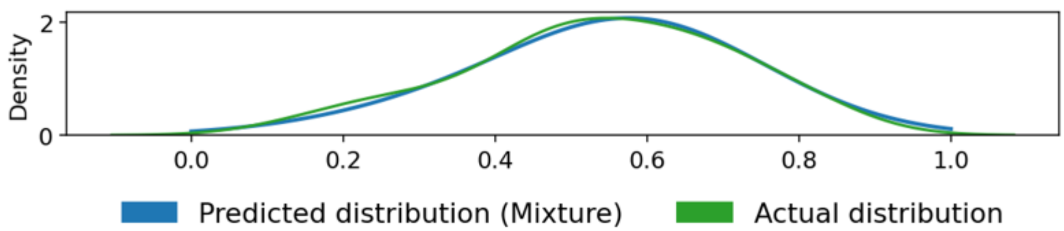

Figure 2-2 Traffic density versus predicted and actual distributions

This was novel machine learning work, and we believed it would yield a simulator that was much more true to the nature of traffic.

In summary, our work included a scheduling queue meso-simulator and machine learning algorithms for learning arrival time distributions.

2.2. Simulation Model Results

The experiments were conducted by using the Python programming language with Python version 3.8.5. We collected data from VISSIM to get data comparable to real-life scenarios.

2.2.1 Statistical Test

The distribution of travel times of vehicles in our simulator was tested against the distribution of travel times of vehicles in VISSIM. For our tests, the null hypothesis was that the travel time distribution of our simulator was the same as the travel time distribution of VISSIM. We used the Mann-Whitney U Test [9] to prove our hypothesis. We obtained a p-value by using the Mann-Whitney U Test. If the p-value was less than 0.05, we rejected the null hypothesis that the two distributions were the same. Otherwise, we failed to reject the null hypothesis that the two distributions were the same. We denoted `Reject' to reject the null hypothesis and `Accept' to fail to reject the null hypothesis.

2.2.2 Road network

We collected data from VISSIM for the road networks shown in figures 2-3 and 2-4. Figure 2-3 shows a road network with a single road segment/edge and two nodes. The route of vehicles had node 1 as the origin and node 2 as the destination, with a single path.

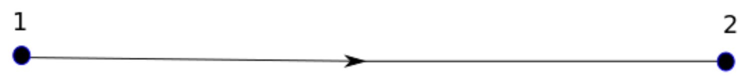

Figure 2-3 Road network with a single road segment

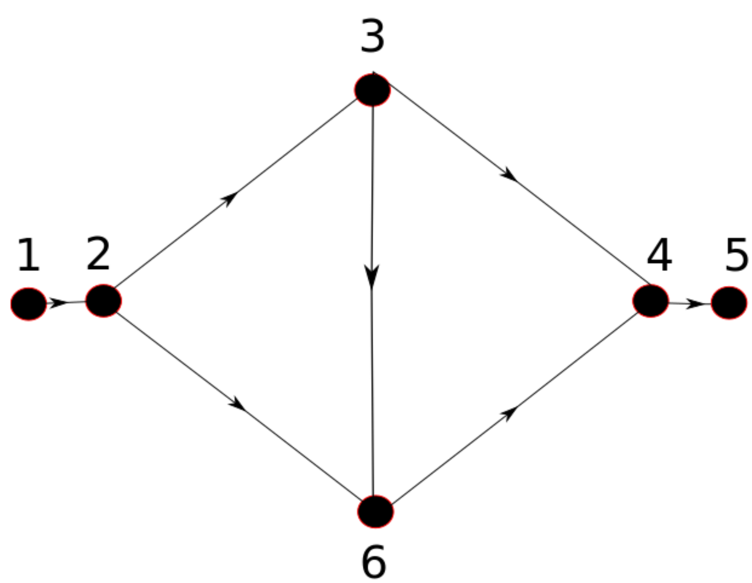

Figure 2-4 Road network with multiple road segments

The road network shown in Figure 2-4 had multiple road segments and six nodes and seven edges. The route of vehicles had node 1 as the origin and node 5 as the destination. Three paths existed between nodes 1 and 5. We tested our simulator against VISSIM for different parameters on these road networks. This is shown in the following sections. These road networks were chosen to test the performance of our simulator for road networks with single and multiple segments. Of the data, 80 percent was used for training (with 10 percent used for validation) and 20 percent was used for testing.

2.2.3 Testing on Different Numbers of Road Segments and Lanes

In this work, a "single segment" refers to a road network with a single road segment, as shown in Figure 2-3. "Multiple segment" refers to a road network with multiple road segments, as shown in Figure 2-4. We set the vehicle input rate to 100 vehicles per hour.

The null hypothesis was that the travel time (total) distribution of our simulator and VISSIM were the same. We denoted `Reject' to reject the null hypothesis and `Accept' to fail to reject the null hypothesis. In Table 2-1 we can see that in all the cases we achieved `Accept,' indicating that our simulator worked the same way as VISSIM for different numbers of road segments and lanes.

Table 2-1 Testing our simulator against VISSIM for different numbers of road segments and lanes.

Num. of road segments	Num. of Lanes	P-value	Reject/Accept
Single	1	0.95	Accept
Single	2	0.93	Accept
Single	3	0.97	Accept
Multiple	1	0.77	Accept
Multiple	2	0.73	Accept
Multiple	3	0.73	Accept

We tested our simulator against VISSIM for road networks with single and multiple road segments. We also tested for one, two, and three lanes for each segment. The distribution of travel times in our simulator were similar to those of VISSIM for different numbers of road segments and lanes. This is shown in Table 2-1. In all the cases, we achieved `Accept,' indicating that our simulator worked the same way as VISSIM for single and multiple road segments and for different numbers of lanes. Note that the p-values for the single segment road networks were similar whether the networks had one, two, or three lanes. Also, the p-values for the multiple segment road networks were similar whether the networks had one, two, or three lanes.

2.2.4 Testing on Different Congestion Levels/Vehicle Input Rates

The driving behavior of an individual varies with congestion on the road. The congestion during the daytime is different from that in the early morning and at night. Congestion may be higher when people leave from major events or if they are going to or from work. At other times, say late night or noon, congestion may be lower. We tested our simulator against VISSIM for different congestion levels/ vehicle input rates. Testing was performed on a multiple-segment road network (Figure 2-4) with three lanes. The distribution of travel times in our simulator was tested against the distribution of travel times in VISSIM by using the Mann-Whitney U Test [9].

The null hypothesis was that the travel time (total) distribution of our simulator and VISSIM were the same. We denoted `Reject' to reject the null hypothesis and `Accept' to fail to reject the null hypothesis. We compared the distributions of total travel time for the vehicles in our simulator and VISSIM. Table 2-2 shows the p-values and Reject/Accept values for different scenarios. In all cases, we achieved `Accept,' indicating that our simulator worked the same way as VISSIM for different congestion levels.

Table 2-2. Testing our simulator against VISSIM for different congestion levels.

Congestion (Vehicles/hr)	P-value	Reject/Accept
10	0.93	Accept
50	0.59	Accept
100	0.73	Accept
500	0.87	Accept

1500	0.68	Accept
1800	0.52	Accept
2100	0.61	Accept

2.2.5 Testing on Different Speed Limits

The driving behavior of an individual varies with the speed limits of the road. We set the vehicle input rate to 100 vehicles per hour and tested our simulator against VISSIM for different speed limits. The distribution of travel times in our simulator was tested against the distribution of travel times in VISSIM by using the Mann-Whitney U Test [9].

The null hypothesis was that the travel time (total) distribution of our simulator and that of VISSIM were the same. We denoted `Reject' to reject the null hypothesis and `Accept' to fail to reject the null hypothesis. Table 2-3 shows the p-values and Reject/Accept values for different scenarios. Testing was done on a multiple-segment road network (Figure 2-4) with a single lane. We compared the distributions of total travel time for the vehicles in our simulator and VISSIM. Table 2-3 shows that in all the cases, we achieved `Accept,' indicating that our simulator worked the same way as VISSIM for different speed limits.

Table 2-3 Testing our simulator against VISSIM for different speed limits.

Speed limit (km/hr)	P-value	Reject/Accept
20	0.79	Accept
30	0.78	Accept
40	0.84	Accept
60	0.76	Accept
70	0.79	Accept

2.2.6 Testing on Road Segments That Are Uphill or Downhill

The driving behavior of an individual uphill is different from the driving behavior downhill. We performed testing on uphill and downhill road segments. The uphill road segments had a positive slope/gradient, and the downhill road segments had a negative slope/gradient. We set the vehicle input rate to 100 vehicles per hour. We tested our simulator against VISSIM for different gradient/slope values. The distribution of travel times in our simulator was tested against the distribution of travel times in VISSIM by using the Mann-Whitney U Test [9].

The null hypothesis was that the travel time (total) distribution of our simulator and VISSIM were the same. We denoted `Reject' to reject the null hypothesis and `Accept' to fail to reject the null hypothesis. Table 2-4 shows the p-values and Reject/Accept values for different scenarios. Testing was done on a multiple-segment road network (Figure. 2-4) with a single lane. We compared the distributions of total travel time for the vehicles in our simulator and VISSIM. Table 2-4 shows that in all the cases, we achieved `Accept,' indicating that our simulator worked the same way as VISSIM for different gradient/slope values.

Table 2-4 Testing our simulator against VISSIM for different gradient/slope values.

Gradient/Slope values	P-value	Reject/Accept
0%	0.73	Accept
-10%, 0%, 10%	0.94	Accept
-20%, 0%, 20%	0.73	Accept
-100%, 0%, 100%	0.89	Accept
-200%, -100%, 0%, 100%,200%	0.78	Accept
-20%, -10%, 0%, 10%,20%	0.86	Accept

2.2.7 Running Time Comparison

Figure 2-5 shows a box and whisker plot of the running times (seconds) of simulations performed on our simulator and VISSIM. The plot shows the running times of 72 simulations and that our simulator had a significantly lower running time than VISSIM. The median, minimum, and maximum running times in VISSIM were 492.74 s, 452.47 s, and 593.58 s, respectively. The median, minimum, and maximum running times in our simulator were 29.18 s, 26.86 s, and 57.34 s, respectively. The ratio of the running time of simulations in VISSIM and our simulator had a range of 10.02 to 21.45. In other words, our simulator ran 10.02 to 21.45 times faster than VISSIM. Note that we used the Python programming language, and Python is 500 times slower than C. Nevertheless, our simulator was still at least 10.02 times faster.

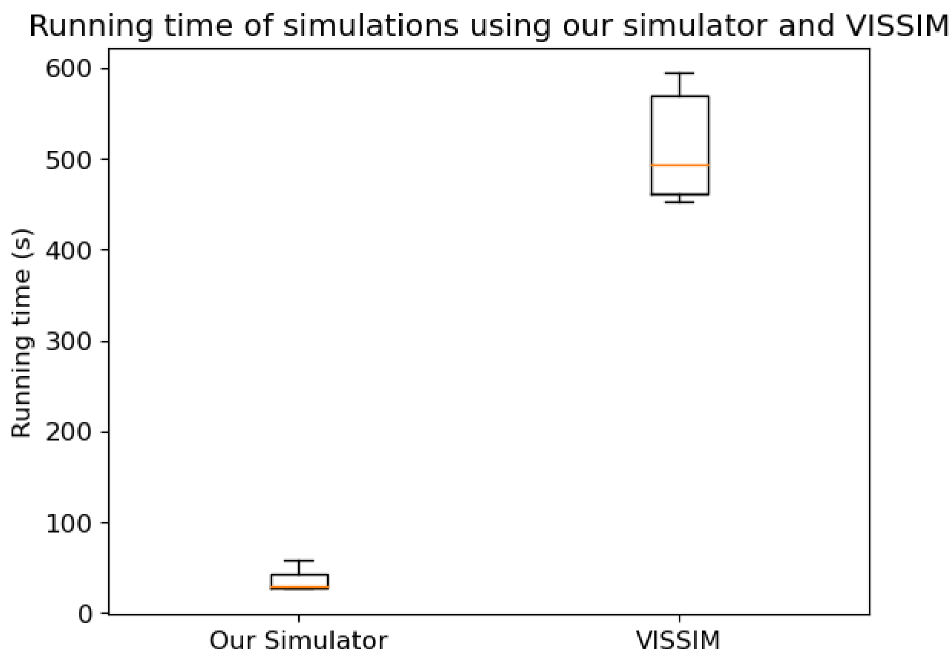

Figure 2-5 Box and whisker plot of running times (s) of simulations using our simulator and VISSIM.

CHAPTER 3: CONCLUSION

We developed a mesoscopic simulator that has a predictor for each road segment. The predictor for a road segment predicts the travel time distribution for the particular traffic conditions. The mesoscopic simulator selects a random sample of the predicted distribution as travel time. We compared the performance of our simulator with that of VISSIM, which stood in for real-time traffic data.

We demonstrated through our experiments that the travel time distributions predicted by our simulator were similar to those of VISSIM for various scenarios. We also conducted experiments to test the running times of simulations using our simulator and VISSIM, and we found that our simulator ran 10.02 to 21.45 times faster than VISSIM.

REFERENCES

[1] Zainab Abbas, Ahmad Al-Shishtawy, Sarunas Girdzijauskas, and Vladimir Vlassov. Short-term traffic pre- diction using long short-term memory neural networks. In 2018 IEEE International Congress on Big Data (BigData Congress), pages 57–65. IEEE, 2018.

[2] Walaa Alajali, Wei Zhou, Sheng Wen, and Yu Wang. Intersection traffic prediction using decision tree models. Symmetry, 10(9):386, 2018.

[3] A Ata, Muhammad Adnan Khan, Sagheer Abbas, Gulzar Ahmad, and Areej Fatima. Modelling smart road traffic congestion control system using machine learning techniques. Neural Network World, 29(2):99–110, 2019.

[4] Nejdet Dogru and Abdulhamit Subasi. Traffic accident detection by using machine learning methods. In Third International Symposium on Sustainable Development (ISSD12), page 467, 2012.

[5] Christoph Ide, Fabian Hadiji, Lars Habel, Alejandro Molina, Thomas Zaksek, Michael Schreckenberg, Kris- tian Kersting, and Christian Wietfeld. Lte connectivity and vehicular traffic prediction based on machine learning approaches. In 2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall), pages 1–5. IEEE, 2015.

[6] Yunxiang Liu and Hao Wu. Prediction of road traffic congestion based on random forest. In 2017 10th International Symposium on Computational Intelligence and Design (ISCID), volume 2, pages 361–364. IEEE, 2017.

[7] PacTrans Board of Directors. 2020 critical research needs, 2020. URL: https://depts.washington.edu/ pactform/wp-content/uploads/2020/07/2020Critical_Research_Needs.pdf.

[8] Fan Zhou, Qing Yang, Ting Zhong, Dajiang Chen, and Ning Zhang. Variational graph neural networks for road traffic prediction in intelligent transportation systems. IEEE Transactions on Industrial Informatics, 2020.

[9] Henry B Mann and Donald R Whitney. On a test of whether one of two random variables is stochastically larger than the other. The annals of mathematical statistics, 1947.