Adaptive K-Nearest neighbor classification using Diffusion Decision Making

Tarun Shimoga
Arizona State University
tshimoga@asu.edu

Anil Kumar Motupalli
Arizona State University
amotupal@asu.edu

Abstract

The aim of this project is to have a comprehensive understanding of the relation between KNN classification in pattern recognition and Diffusion Decision making in cognitive learning as explained in (Noh et al., 2012). The goal is to merge the two techniques to enhance K-Nearest neighbor classification procedure specifically in the amount of resource utilized to make a decision. **We extended this idea to a multi-class voting based classification with pleasing results shown in the last section.**

1 Introduction

K-Nearest neighbor classification is an extension of the Nearest neighbor classification which classifies the Test point x by assigning it the label most frequently represented among the k nearest samples; in other words a decision is made by examining the labels on the k nearest neighbors and taking a vote. More specifically, k-nearest-neighbor rule selects class ω_m if a majority of the k nearest neighbors are labeled ω_m an event of probability,

$$\sum_{i=\frac{(k+1)}{2}}^{k} \binom{k}{i} P(\omega_m|x)^i [1 - P(\omega_m|x)]^{k-1}$$

It is evident that, by selecting a large k, the probability that ω_m will be selected increases(Duda et al., 2012). One of the limitations of k-nearest-neighbor classification technique arises in cases where classification decisions can be made with neighbors less than the specified k. If one were to think of considering more neighbors as extra cost, k-nearest-neighbor technique proves to be a very inflexible method. This is where we intend to use the adaptive properties of Diffusion Decision Making.

Diffusion Decision Making explains the decision making process in humans. It states that, a decision is made when evidence, which is represented as a function of the input, reaches a pre determined confidence level (Ratcliff and McKoon, 2008). This flexibility in decision making is what results in saving significant amount of resources. In real life, resource can be time, in terms of statistical pattern recognition, this could be represented in terms of the amount of samples considered.

We aim to exploit the advantages of both these techniques, namely, intuitiveness and tractability of k-nearest-neighbor classification and flexibility of Diffusion Decision Making.

In section 2 we discuss Diffusion Decision in detail and how confidence levels and evidences can be calculated for adaptive k-nearest-neighbor classification. Section 3 explains detailed methodology for different rules of deriving the confidence levels. Section 4 discusses experiment settings, experiments and results. We propose multi-class enhancement in Section 5.

2 Incorporate Diffusion Decision properties in K-Nearest neighbor technique

2.1 Confidence and Evidence in Diffusion Decision

In the above section, the idea of merging the positives of k-nearest-neighbor and Diffusion Decision techniques to generate a more resource conscious and adaptive decision making technique was proposed. The foundation of this technique depends on how strong is the association between evidence and confidence levels in Diffusion Decision making and k-nearest-neighbor. Diffusion Decision making finds the happy balance between accuracy and resource utilization using the Sequential Probability Ratio Test (SPRT). It is shown that, the conventional majority voting rule is identical to the diffusion decision model when applied to data from two different Poisson processes. This stochastic model for decision making considers diffusion of an evidence in favor of either of two possible choices by continuously accumulating the information. After initial wavering between the two choices, the evidence nally reaches a level of condence where a decision is made(Noh et al., 2012). In the studies of decision making in the lateral inter-parietal (LIP) area of the brain, two Poisson processes are assumed to have rate parameters of either λ_+ and λ_- where we know that $\lambda_+ > \lambda_-$, but exact values are unknown. When it should be determined which Poisson process has the larger rate λ_+ among the two competing Poisson distributions for N_1 and N_2 at times T_1 and T_2 respectively, a sequential probability ratio test (SPRT) with pre determined confidence α can be used to explain confidence levels in a diffusion decision model (Wald and Wolfowitz, 1948).

The confidence levels derived are as follows(Noh et al., 2012),

$|\ N_1$ - $N_2\ |$ = z_N where z_N = $\frac{\log \alpha}{\log (\lambda_+/\lambda_-)}$, and

$|\ T_1$ - $T_2\ |$ = z_T where z_T = $\frac{\log \alpha}{(\lambda_+ - \lambda_-)}$

These rules can be used to set confidence levels to minimize resource usage for a k-nearest-neighbor classification. The number of neighbors can contribute towards the diffusion of evidence, and classification decision is made after a confidence level is reached.

2.2 Represent training data as Poisson process to determine rate parameter

From (Noh et al., 2012) and (Leonenko et al., 2008), it is shown that when we consider nearest neighbors of an underlying distribution within a constant rate growing hyper-sphere, the distances of the neighbors follow a Poisson distribution. We will need this behavior to incorporate Diffusion Decision principles for calculating the confidence levels. Once we have the rate parameters, these values can plugged in the SPRT derived formulas.The calculation of the rate parameters is explained in the Methodology section.

3 Methodology

This section explains how different confidence levels are calculated after approximating the appearance of neighbors as Poisson processes. From the findings shown in (Noh et al., 2012), each of these rules used towards decision making has trade-offs between accuracy of classification and number of neighbors needed to be accumulated for evidence. We show the actual findings in the Experiment section below.

3.1 Calculate rate parameters of Poisson processes

The rate parameters that we get after approximating the class training samples to Poisson distribution is required to calculate the confidence levels. This is done by first considering a random sample of the class. Now keeping this sample as the center we calculate the rate at which neighbors of this class appear within a constant rate hyper-sphere. The rate parameter is then obtained by taking an average over all the rates at which neighbors appear. This step is performed for each class before further computation of the confidence levels.

3.2 DN Criteria

3.2.1 Evidence

For this rule, considering the test point as the center of a certain size hyper-sphere, we consider the difference in the number of neighbors of both the classes as evidence. If N_1 and N_2 represent the number of training samples within the hyper-sphere of class 1 and 2, then the evidence is represented as $\mid N_1$ - $N_2 \mid$. If the evidence doesn't reach the calculated confidence level, the size of the hyper-sphere is increased by some unit and then the evidence is checked again. Once the evidence reaches the confidence level, further accumulation of neighbors is stopped and a decision is made.

3.2.2 Confidence

The confidence level for the DN rule is calculated using the formula $\frac{\log \alpha}{\log (\lambda_+/\lambda_-)}$. The components in the above formula are :
1. α - Confidence level set in the SPRT. We will further find out how this affects accuracy and resource usage
2. λ_+ - Rate parameter of the higher rate Poisson process
3. λ_- - Rate parameter of the slower rate Poisson process

3.3 DV Criteria

3.3.1 Evidence

Here we consider the test point and the difference in the volumes with N^{th} nearest neighbor on surface of the hyper-sphere. This evidence is calculated as $\mid V_1$ - $V_2 \mid$.

3.3.2 Confidence

The confidence level for the DV rule is calculated using the formula $\frac{\log \alpha}{(\lambda_+ - \lambda_-)}$.
The components in the above formula are :
1. α - Confidence level set in the SPRT.
2. λ_+ - Rate parameter of the higher rate Poisson process
3. λ_- - Rate parameter of the slower rate Poisson process

3.4 CDV Criteria

3.4.1 Evidence

The principle for this rule is very similar to the DV rule apart from the fact that it is more conservative in the nearest neighbor considered. Here instead we consider the test point and its distance to the $(N+1)^{th}$ nearest neighbor of the second class.This results in the difference to be smaller which eventually needs more neighbors to reach a confidence. If V_1 and V_2 are represented as the volumes of the hyper-spheres respectively, the evidence is $\mid V_1$ - $V_2 \mid$.

3.4.2 Confidence

The confidence level for the CDV rule is calculated using the formula $\frac{\log \alpha}{(\lambda_+ - \lambda_-)}$.
The components in the above formula are :
1. α - Confidence level set in the SPRT.
2. λ_+ - Rate parameter of the higher rate Poisson process
3. λ_- - Rate parameter of the slower rate Poisson process

3.5 Bayesian Criteria

For the Bayesian Criteria, we assume the rate parameters to be random variables. This gives rise to two forms of sub-criteria as explained below. We calculate the probability of p($\lambda_+ > \lambda_-$) by using the number of nearest neighbors and the volume of the hyper-sphere generated by the N^{th} nearest neighbor. Is is divided into two rules one based on the parameter we consider to calculate

the probability p($\lambda_+ > \lambda_-$).
Starting with a conjugate prior defined as p(λ) = $\frac{b^a}{\Gamma(a)}$exp(-λb), posterior probabilities are calculated for the two cases mentioned below

3.5.1 PDV Criteria (using N^{th}earest neighbors to calculate P($\lambda_+ > \lambda_-$)

Evidence

Evidence, P($\lambda_1 > \lambda_2 | u_1, u_2$) is calculated as follows (Noh et al., 2012):

$$\sum_{m=0}^{N+a-1} \binom{2N+2a-1}{m} \frac{(u_1+b)^m (u_2+b)^{2N+2a-1-m}}{(u_1+u_2+2b)^{2N+2a-1}}$$

Confidence

Confidence levels are determined by performing multiple experiments and evaluating the evidence vs accuracy trade-off. In our experiment, we tested with confidence levels of 0.51,0.6,0.7 and 0.8.

3.6 PDN Criteria (using neighbors within constantly increasing hyper-sphere to calculate P($\lambda_+ > \lambda_-$)

Evidence

Evidence, P($\lambda_1 > \lambda_2 | N_1, N_2$) is calculated as follows (Noh et al., 2012):

$$\frac{1}{2^{N_1+N_2+2a-1}} \sum_{m=0}^{N_1+a-1} \binom{N_1+N_2+2a-1}{m}$$

Confidence

Confidence levels are determined by performing multiple experiments and evaluating the evidence vs accuracy trade-off. In our experiment, we tested with confidence levels of 0.51,0.55,0.6 and 0.65.

4 Experiments

In this section we empirically evaluate classification based on the confidence levels determined from the above rules.

4.1 Fisher Discriminant for dimension reduction

We start the experiment by first reducing the 3072 dimension raw data to 9 dimension space.

4.2 DN Criteria

We ran the experiment for different confidence values of α. Below is a table that shows the comparison for number of neighbors and accuracy.

Confidence level α	Average Number of neighbors	Accuracy
1.1	5.46	74.44
3	46.87	76.97
5	63.09	76.83
7	72.72	76.77
10	82.53	76.95

Table 1: Classification accuracy and number of resources for DN rule with different confidence levels

The graph below shows a comparison in accuracy for K-Nearest Neighbor classification against DN criteria.

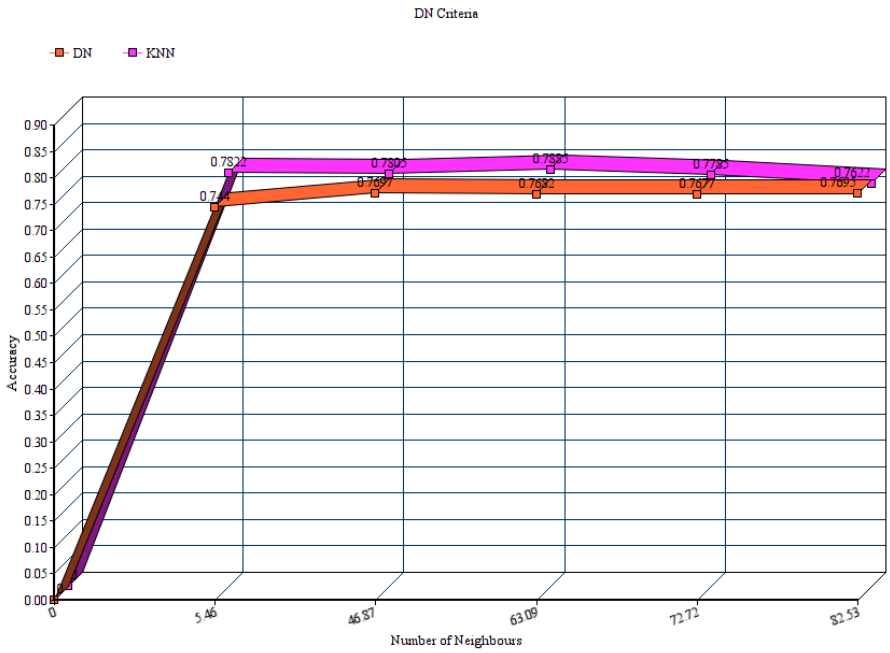

Figure 1: Number of neighbors vs Accuracy comparison of KNN classification and DN criteria

4.3 DV Criteria

Confidence level to accuracy and number of neighbors relation for DV criteria is shown in the table below

Confidence level α	Average Number of neighbors	Accuracy in %
1.1	2.016	71.97
3	2.73	72.84
5	3.322	72.93

Table 2: Classification accuracy and number of resources for DV rule with different confidence levels

The graph below shows a comparison in accuracy for K-Nearest Neighbor classification against DV criteria.

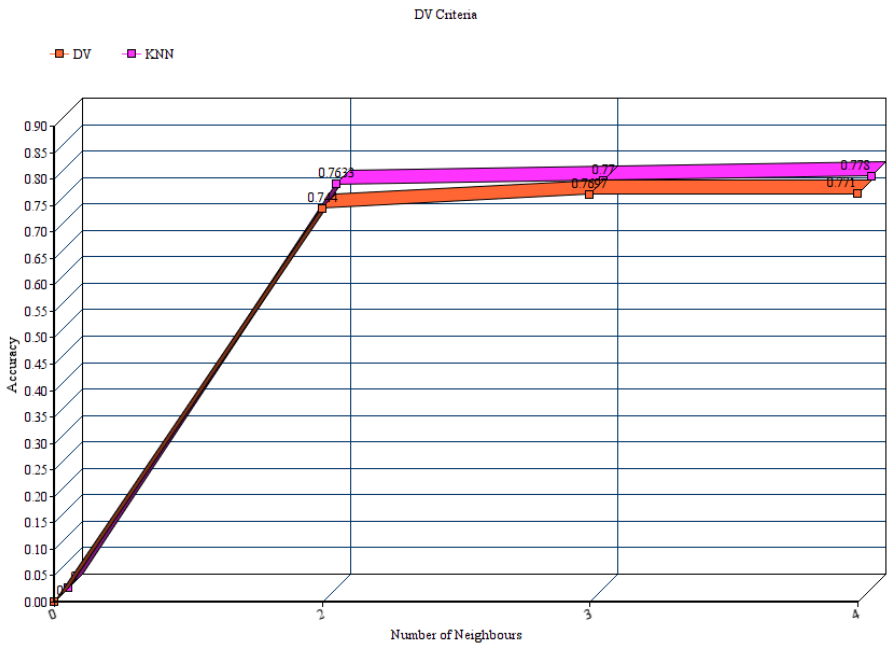

Figure 2: Number of neighbors vs Accuracy comparison of KNN classification and DV criteria

4.4 CDV Criteria

Confidence level to accuracy and number of neighbors relation for the Conservative DV criteria is shown in the table below

Confidence level α	Average Number of neighbor	Accuracy in %
1.1	2.012	71.8
3	2.72	72.68
4	3.017	72.95
5	3.239	73.2

Table 3: Classification accuracy and number of resources for CDV rule with different confidence levels

The graph below shows a comparison in accuracy for K-Nearest Neighbor classification against CDV criteria.

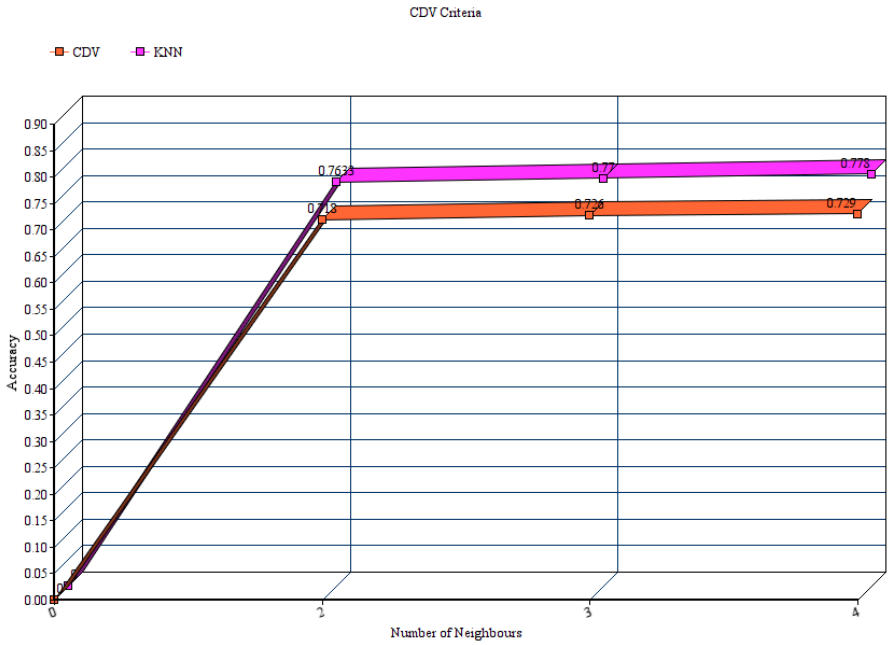

Figure 3: Number of neighbors vs Accuracy comparison of KNN classification and CDV criteria

4.5 Bayes Criteria

4.5.1 PDV Criteria

For the PDV criteria, the confidence levels to accuracy and confidence level to number of neighbors relation is shown in the table below

Confidence level	Average Number of neighbors	Accuracy in %
0.51	2.036	71.9
0.55	2.249	73.1
0.60	2.809	74.3
0.65	4.024	75.4

Table 4: Classification accuracy and number of resources for PDV rule with different confidence levels

The graph below shows a comparison in accuracy for K-Nearest Neighbor classification against PDV criteria.

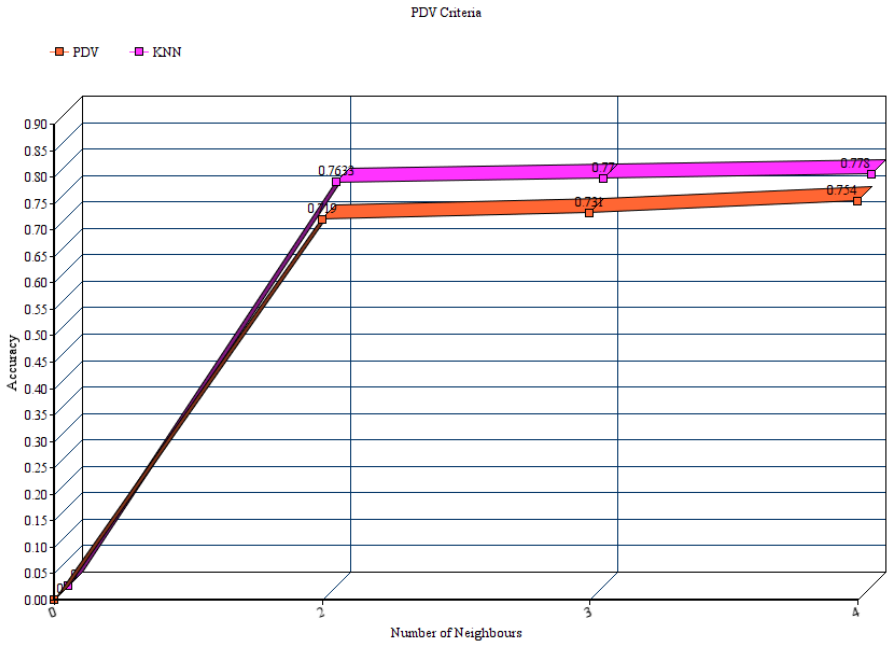

Figure 4: Number of neighbors vs Accuracy comparison of KNN classification and PDV criteria

4.5.2 PDN Criteria

For the PDN criteria, the confidence levels to accuracy and confidence level to number of neighbors relation is shown in the table below

Confidence level	Average Number of neighbors	Accuracy in %
0.51	2.199	73.73
0.55	2.199	73.73
0.60	2.20	73.73
0.65	2.313	73.77
0.70	3.12	74.33
0.80	10.37	76.24

Table 5: Classification accuracy and number of resources for PDN rule with different confidence levels

The graph below shows a comparison in accuracy for K-Nearest Neighbor classification against PDN criteria.

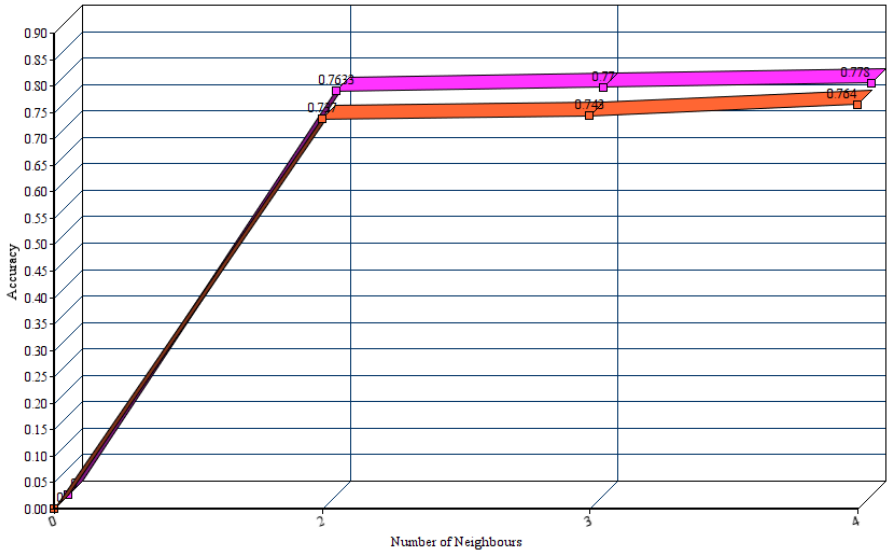

Figure 5: Number of neighbors vs Accuracy comparison of KNN classification and PDN criteria

5 Extension to Multi-class classification

This binary classification concept can be extended to a multi-class classification wherein each test point is checked for $\binom{N}{2}$ combinations and assign the label of the highest vote class. In our experiments, each test point was checked for $\binom{10}{2}$ combinations giving rise to a matrix of labels. The test point is then assigned to the most occurring label.

The table below shows the accuracy comparison between KNN and DN criteria

Number of Neighbors	KNN accuracy in %	Adaptive KNN (DN criteria) accuracy in %
1068	37.08	35.68 ($\alpha = 1.1$)
1078	37.00	37.00 ($\alpha = 2$)

Table 6: Comparison of classification accuracy and number of neighbors for Multi-class classification for KNN versus Adaptive KNN

6 Conclusion

Diffusion Decision model gives comparable results to KNN and sometimes better results with a proper confidence value. As we increase the confidence value, accuracy increases but the program struggles to come to a decision. We can extend this algorithm to multi-class classification, but the complexity increases as the program takes ${}^{n}C_{d}$ time. We had a good learning experience from the project. We have learned the basics of how to approach a classification problem, how to solve it, which tools to use and where to find the help.

7 Efforts Distribution

Team Member	Coding	Report
Anil Motupalli	60%	40%
Tarun Shimoga	40%	60%

Table 7: Efforts distribution in percentage

8 Instructions to run the program

We are using anaconda distribution of python for this project because of it's flexibility to install new packages and it is a specifically made for machine learning. We used scikit-learn(Pedregosa et al., 2011), numpy, math, sys, scipy, pickle, cpickle, decimal libraries for this project. Please make sure these libraries available in your version of python before running the code. Make sure the directory you place the code has the write access to that directory.

1.Run data_pickler.py file to perform FDA and pickle the data into .pickle file.
2.Run generate_lamdas.py file to generate lamdas for different classes and it'll be pickled into .pickle file.
3.Run the corresponding file for the rule as specified below:-

a. DN criteria	deltaN.py
b. DV criteria	deltaV.py
c. CDV criteria	CDV.py
d. PDN criteria	bayesian_wth_N.py
e. PDV criteria	bayesian_with_U.py
f. Multi-class classification using DN	deltaN_multiclas.py
g. K Nearest Neighbor(binary)	KNN.py
h. K Nearest Neighbor(multi class)	KNN_multiclass.py

References

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern classification*. John Wiley & Sons, 2012.

N. Leonenko, L. Pronzato, V. Savani, et al. A class of rényi information estimators for multidimensional densities. *The Annals of Statistics*, 36(5):2153–2182, 2008.

Y.-K. Noh, F. Park, and D. D. Lee. Diffusion decision making for adaptive k-nearest neighbor classification. In *Advances in Neural Information Processing Systems*, pages 1925–1933, 2012.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, 12:2825–2830, 2011.

R. Ratcliff and G. McKoon. The diffusion decision model: theory and data for two-choice decision tasks. *Neural computation*, 20(4):873–922, 2008.

A. Wald and J. Wolfowitz. Optimum character of the sequential probability ratio test. *The Annals of Mathematical Statistics*, pages 326–339, 1948.