Monitoring and diagnosis of processes in which their transactions fall in multiple categories

RODRIGO I. DURAN* and SUSAN L. ALBIN

Department of Industrial and Systems Engineering

Rutgers University

96 Frelinghuysen Road, Piscataway, New Jersey 08854, USA

E-mail: rodduran@eden.rutgers.edu, salbin@rci.rutgers.edu

Abstract

We consider a process where transactions, such as customer service transactions, are classified into categories. With just two categories, the fractions in each can be monitored with the familiar *p-chart* based on the Binomial distribution. With more than two categories, the fractions can be monitored with a method based on a test statistics as in Marcucci (1985). Here we present a new method for monitoring the fraction of transactions among categories called the *p-tree*. We show that any multinomial with *K* categories can be transformed into a probability tree with *K*-1 binary stages, and monitored with *K*-1 independent *p-chart*s. In contrast to Marcucci's method, the *p-tree* method provides an accurate and easy way to interpret signals by helping to pinpoint the category where there has been a disturbance. There is no limit on the number of categories. Simulation studies show the *p-tree* method is a helpful diagnostic tool while the Marcucci's method is not. The simulations also show that the *p-tree* and Marcucci methods have comparable sensitivity.

* Corresponding author:

1 Introduction

Consider a process where transactions are classified in several categories. The transactions could be related to customer service or to manufactured products; however, the primary interest here is in the customer service area. The work was motivated by the previous experience of the first author who served as director for the national administration of the property tax in Chile.

Consider how categories are formed in a property tax service noting that the system has many similarities to private sector service systems. Notices are sent monthly in batches to the taxpayers to communicate changes in the assessment. The taxpayers have one month to file a complaint. Rafool (2002) contains an overview of the US property tax system and The New Jersey Property Tax Assessment Study Commission (1986) describes methods and makes recommendations, which are still valid, about the administration of this tax in New Jersey.

The assessment process generates errors that the taxpayer and the property tax system must resolve. A taxpayer with a problem passes through two stages: the first is to consult with a front desk assessor and the second is to actually file a complaint (assuming that first stage is always required before filing a complaint). The management, at the end of the deadline, may classify the sent notices into three categories: not consulted, consulted but not filed, and consulted and filed complaints

When there are only two categories, the fractions are monitored with the familiar *p-chart* based on the Binomial distribution for the realized numbers in one of the two categories. With more than two categories, these fractions can be monitored with a control chart based on a test statistics as in Marcucci (1985). This method assumes that the probability parameters do not depend on the sample volume and samples are not autocorrelated over time. Marcucci's method is simple to use and detects increases and decreases. However, it is difficult to interpret an out-of-control signal.

Marcucci (1985) gives data where samples of finished brick are classified into conforming, nonconforming type *A*, and nonconforming type *B* categories with baseline probabilities 0.95, 0.03, and 0.02 respectively. Table 1 shows simulated data using the Marcucci brick example parameters.

	Fraction		
	conforming	*non-conforming A*	*non-conforming B*
Baseline	.950	.030	.020
Obs 1	.960	.014	.026
Obs 2	.932	.034	.034

Table 1. Marcucci's method for two out-of-control observations

Table 1 demonstrates that is difficult to interpret the results. Both observations are significantly outside the control limit with a false alarm rate of 0.05 as used by Marcucci (1985). For observation 1, is the out-of-control condition caused by an increase in the number of conforming bricks? For observation 2, does the proportion of nonconforming type *A* and nonconforming type *B* bricks cause the out-of-control condition? It is difficult to make a conclusion by eye since as one count increases then the sum of the other two decreases, and vice-versa; i.e., the numbers are negatively correlated. Indeed, Simonoff (2003, p. 80) explains that the test statistic used by Marcucci, i.e., the Pearson statistic, cannot detect in which categories or in which direction the shift is occurring.

Marcucci (1985) attempts interpretation as follows: two out of the three counts are selected and then modified *p-charts* are created. However, it is not clear which count should be discarded. Marcucci (1985, p. 89) indicates that this method is restricted to at most three categories.

Here we present a new method that offers an easier way to interpret an out-of-control signal. The new method is called the *p-tree* method in that any multinomial process with *K* categories is transformed into a probability tree with *K*-1 binary stages. The tree fractions at every stage are independent, as based on Johnson *et al.* (1996, p. 68) as well as Kemp and Kemp (1987), and can be monitored with independent *p-chart*s. Thus, the *p-tree* method indicates easily which stages are responsible in case of a signal.

For example, the *p-tree* method applied to the observations in Table 1, indicates that in observation 1 the fraction conforming brick is consistent with baseline, but among nonconforming, type *A* is underrepresented. For observation 2, the *p-tree* method indicates that the fraction conforming is low compared to baseline and the fraction of type *A* among nonconforming is consistent with baseline. Applying the *p-tree* method to the taxpayers' process, we first monitor the fraction of taxpayers that consult in a month with the stage-one *p-chart*. Then we monitor the fraction of filed complaints over the number of taxpayers that consult with the stage-two *p-chart*.

Monitoring customer transaction processes can be somewhat different from monitoring manufacturing processes. Customer transactions processes are usually continuously scanned through an IT system, in contrast to manufacturing processes where sampling is often used. In customer transactions the monitoring method might continue to provide signals after its first signal because it may not be possible to remove special causes to return the process quickly to in-control. For example, if the *p-tree* method shows that the fraction of complaints in the tax assessment is abnormally large, then management may introduce corrective actions like better regulations, improved instructions, and changes in the IT system – but such changes may take some time.

We propose to measure the diagnosis accuracy of the *p-tree* method by comparing the number of correct signals with all signals in simulation experiments. We also measure the average run length (ARL), which is an estimate of the expected number of samples in a control chart until a sample indicates an out-of-control, given the process is out-of-control. The simulation studies show that the *p-tree* method is a helpful diagnostic tool where the Marcucci's method offers no diagnostic help. Also, the simulation studies show that the *p-tree* method has about the same sensitivity as the Marcucci's method when comparing ARLs. Further, the *p-tree*'s diagnosis features appear to work well regardless of the number of categories.

Another approach for monitoring a multinomial process is Bayesian, as shown in Laviolette (1995), and Shiau *et al.* (2005). In general, the Bayesian approach adds the assumption that the probability parameters of the multinomial model vary according to a prior distribution, typically the Dirichlet distribution. Neither Bayesian method aids in interpretation of an out-of-control signal. Further, Laviolette (1995) detects only increases in nonconforming fractions in contrast with the *p-tree* method which is two-sided, i.e., detects both increases and decreases of fractions.

The limitation in Shiau *et al.* (2005) is that they propose a separate chart for each fraction without taking into account that fractions in every category are negatively correlated. In case of the bricks example, their method would monitor the fractions in Table 1 with three separate *p-charts*. The recommended control limits in Shiau *et al.* (2005) are of the Bonferroni type and lead to a false alarm rate of the combined three *p-chart*s not greater than the desired total false alarm rate. However, using separate charts for correlated variables produces misleading results as noted in Montgomery (2005, p. 487-488 and p. 499) and Lowry *et al.* (1992, p. 52). Indeed, separate charts may have an excessively large ARL or may not signal at all for samples that are out-of-control. The data in Shiau *et al.* (2005, p. 25-27) even produces contradictions where a chart signals a significant increase in one category while no other category chart signals a significant decrease. These criticisms do not apply to the *p-tree* method because it uses independent fractions. Additionally, Shiau *et al.* (2005) propose randomized control limits which are cumbersome in practice; i.e., those samples that fall on control limits are randomly assigned to in-control or out-of-control.

Tucker *et al.* (2002) proposes a monitoring method for ordinal categorical data. The method assumes that the ordinal characteristic has an underlying continuous distribution. If the user guesses the underlying distribution correctly, then the sensitivity of the ordinal chart is better that the Marcucci's method. The method in Tucker *et al.* is not applicable to nominal categorical data. No interpretation of signals is provided. Both the *p-tree* and Marcucci methods may monitor either nominal or ordinal categorical data.

The rest of this paper is organized as follows: Section 2 gives the equivalence between a multinomial process and a probability tree. Section 3 describes the *p-tree* method's control charts based on the independence of the tree fractions. Section 4 shows simulation results to illustrate the diagnosis capabilities of the *p-tree* method and compares its ARL performance with the multinomial based chart of Marcucci's method.

2 Equivalence between multinomial process and probability tree

Consider a multinomial with 3 categories (trinomial) and a sample of size N transactions. Baseline probabilities in category i are equal to p_i, i= 1,2,3, and the realized numbers in each category are n_i, i = 1, 2, 3. Of course, $p_1 + p_2 + p_3 = 1$ and $n_1 + n_2 + n_3 = N$. This process is depicted in Figure 1a. Equivalently, Figure1b depicts this trinomial process as a probability tree with 2 stages. In stage 1, f_1 is the baseline probability that transactions are in category 1 resulting in realized number n_1, and $1\text{-}f_1$ is the probability that transactions are not in category 1, with realized number $N\text{-}n_1$. In stage 2, f_2 is the baseline conditional probability that transactions are in category 2 given that they are not in category 1, with realized number n_2. The conditional probability that transactions are not in category 2 (therefore in category 3) given that they are not in category 1 is $1\text{-}f_2$, with realized number n_3. Only two conditional probability parameters completely characterize the process, f_1 and f_2.

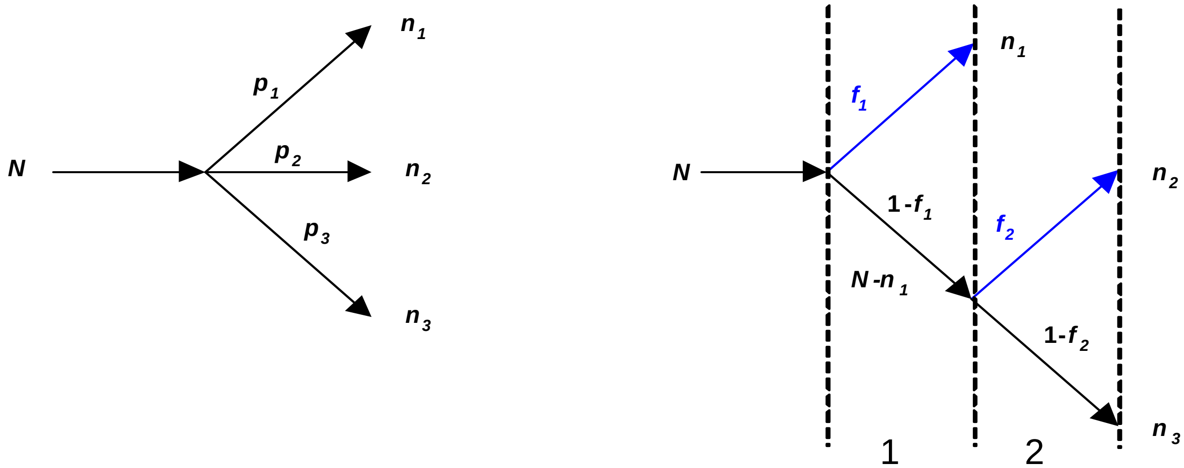

Figure 1. (a) A trinomial process and

(b) Trinomial process represented as a probability tree with 2 stages.

Based on the probability multiplicative rule, the p_i may be expressed as a function of the f_i: $p_1 = f_1$, $p_2 = (1 - f_1) \cdot f_2$, $p_3 = (1 - f_1) \cdot (1 - f_2)$. It follows that $f_2 = p_2 / (1 - p_1)$.

This probability tree representation allows the user to order the categories according to their monitoring importance. In the Marcucci bricks example it is logical to order the stages such that stage 1 discriminates between conforming and non-conforming bricks, with f_1 equal to the fraction of conforming bricks, and stage 2 discriminates between non-conforming Type *A* and *B* bricks with f_2 the conditional probability of non-conforming Type *A* given non-conforming. In case of the categories are of equal or unknown importance, such as transactions type *A*, *B*, or *C*, one can choose an order by default, like categories in decreasing order of their in-control probabilities p_i.

In general, any multinomial with *K* categories can be monitored by a *p-tree* method that consists of *K*-1 independent *p*-control charts, as shown in next section. For the probability tree, f_i = baseline conditional probability that item is in category *i* given that is not in category 1,…,

i-1 for $i \geq 2$. The relationships between the multinomial and tree parameters are based on the total probability rule for multiple events as shown in Montgomery and Runger (2002, p. 44-45):

$$p_1 = f_1 \qquad \text{and} \qquad p_i = [\prod_{j=1}^{i-1}(1-f_j)] \cdot f_i \qquad \text{for } i=2,\ldots,K$$

It follows that,

$$f_i = \frac{p_i}{1-\sum_{j=1}^{i-1} p_j} \qquad i=2,\ldots,K \qquad (1)$$

A shift in a probability p_j produces shifts in every tree probability f_i with $i \geq j$. Similarly, a shift in a tree probability f_i may have been produced by a shift on any probability p_j with $j \leq i$. The last fraction f_K always equals one, meaning that if the item is not in categories 1,2,…,K-1, it must be in category K.

3 The *p-tree* method control charts

The *p-tree* method monitors any multinomial process with K-1 control charts that are independent as discussed below. Assuming that the baseline probabilities p_i are known, then according to (1) the baseline conditional probabilities f_i, i=1,…,K-1 are also known. The control limits are:

$$\begin{cases} UCL_1 = f_1 + Z_{(1-\alpha^*/2)} \cdot \sqrt{\dfrac{f_1(1-f_1)}{N}} \\ CL_1 = f_1 \\ LCL_1 = f_1 - Z_{(1-\alpha^*/2)} \cdot \sqrt{\dfrac{f_1(1-f_1)}{N}} \end{cases} \qquad (2)$$

$$\begin{cases} UCL_i = f_i + Z_{(1-\alpha^*/2)} \cdot \sqrt{\dfrac{f_i(1-f_i)}{N-\sum_{j=1}^{i-1} n_j}} \\ CL_i = f_i \\ LCL_i = f_i - Z_{(1-\alpha^*/2)} \cdot \sqrt{\dfrac{f_i(1-f_i)}{N-\sum_{j=1}^{i-1} n_j}} \end{cases} \qquad \text{for } i=2,\ldots,K\text{-}1,$$

where Z_p is the standard normal distribution such that the upper tail area is p.

The total false alarm rate for all the charts is α and the false alarm rate for each chart is α* given by:

$$\alpha^* = 1-(1-\alpha)^{\left(\frac{1}{K-1}\right)} \tag{3}$$

based on the independence of the charts (as shown below) and using Montgomery (2005, eqns. (10-1) and (10-2), p. 489).

The sample statistic for each control chart is

$$\hat{f}_1 = \frac{n_1}{N} \quad \text{and} \quad \hat{f}_i = \frac{n_i}{N-\sum_{j=1}^{i-1} n_j} \qquad i=2,\ldots,K\text{-}1 \tag{4}$$

At any observation time, the process is in-control if all *K-1 p-chart*s have sample statistics within the control limits. The process is out-of-control if any of the *p-chart*s signal, i.e., have sample statistics outside the control limits.

These control charts are based on the distributions for every binary stage of the probability tree. These are independent Binomial distributions of the realized number n_i in category i given the realized numbers in categories 1,…,i-1, as shown by Johnson *et al.* (1996, p. 68) as well as Kemp and Kemp (1987), and given by:

$$\begin{cases} n_1 \sim \text{Binomial}(N,\ p_1) \\ n_i \text{ given } n_1, n_2 \dots n_{i-1} \sim \text{Binomial}(N - \sum_{j=1}^{i-1} n_j,\ \dfrac{p_i}{1 - \sum_{j=1}^{i-1} p_j}) \text{ for } i = 2 \dots K\text{-}1 \end{cases} \quad (5)$$

For instances, n_2 comes from an independent Binomial with sample size N-n_1 and probability $\frac{p_2}{1-p_1}$, which according to eqn. (1) is equivalent to f_2. Thus, every *p-chart* monitors the independent n_i in category i given the realized numbers in categories 1,…,i-1, using the sample tree fraction ($\hat{f}_i$) of stage i, as the ratio of the realized n_i to its sample size in its independent Binomial in eqn. (5).

The square-root terms in eqn. (2) are the standard deviations of the $\hat{f}_i$ conditioned on n_1, n_2,…, n_{i-1}, which can be directly obtained from each Binomial in eqn. (5). It can be shown also that those square-root terms represent an approximation for the unconditional standard deviation of $\hat{f}_i$ when N is large and the probability that any count equal zero is negligible.

The *p-tree* method may be implemented using Minitab or any other software that offers the *p-chart*. For example, Figure 2a shows a *p-chart* for $\hat{f}_1$ and Figure 2b shows a *p-chart* for $\hat{f}_2$ for simulated finished bricks data as in Table 1 with $\alpha = 0.05$.

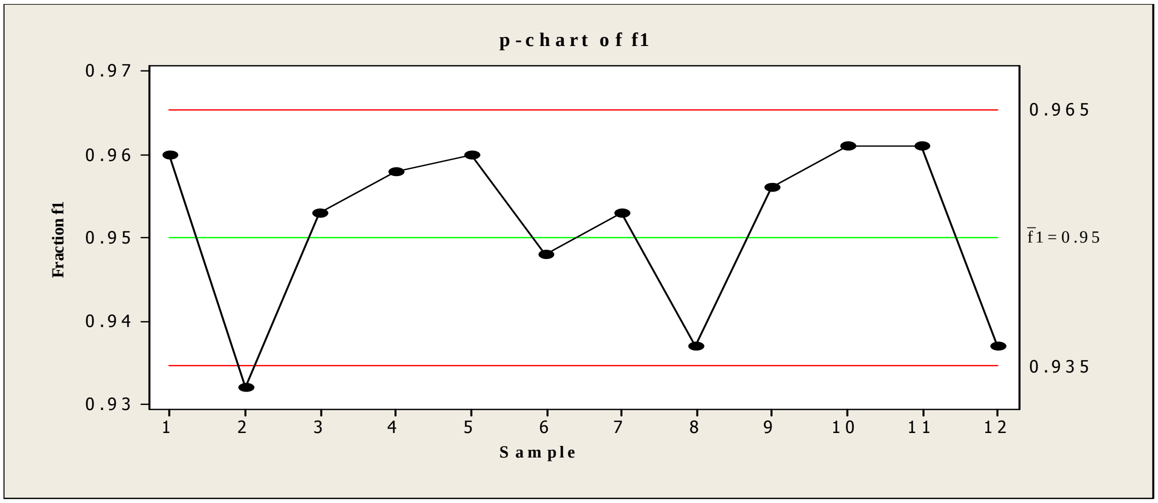

Figure 2a. *p-chart* for f_1 (conforming bricks over total)

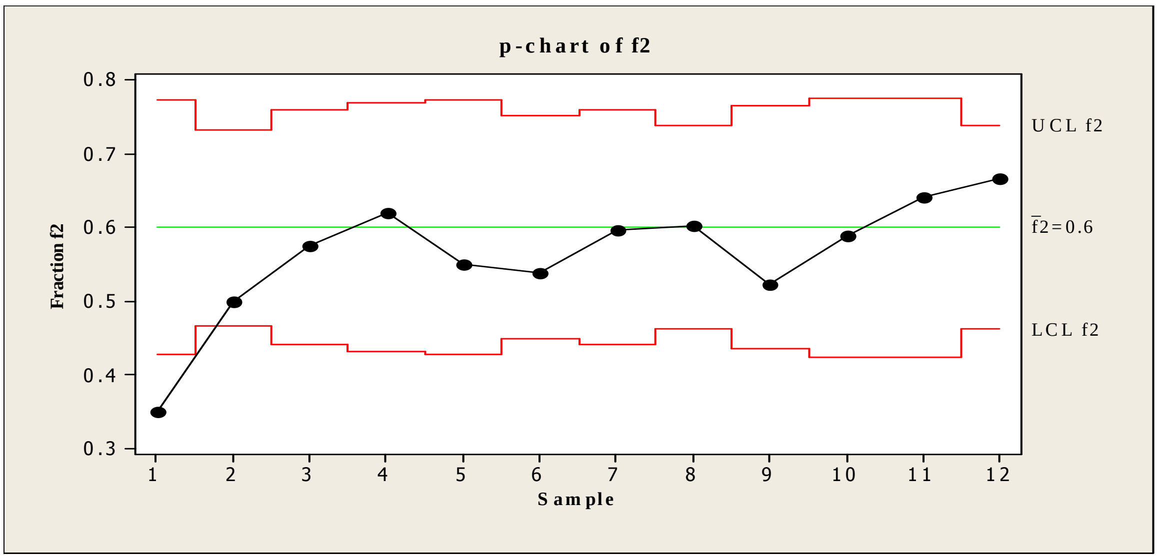

Figure 2b. *p-chart* for f_2 (non-conforming Type *A* bricks over all non-conforming bricks)

The first two observations in Figures 2a and 2b are those in Table 1 and are the only ones outside the control limits. The *p-chart*s make it easy to interpret the out-of-control signals: observation 1 reflects a decrease in type *A* bricks relative to the total non-conforming bricks (as shown in Fig. 2b) while observation 2 reflects a decrease in conforming bricks (as shown in Fig. 2a).

4 Measuring experiment comparing *p-tree* and Marcucci methods

We define the diagnosis accuracy of the *p-tree* method as the fraction of correct signals over the total number of signals, a concept adapted from Skinner *et al.* (2006). If the tree probability f_i shifts, the signal is correct only when the *p-chart* for f_i signals, and other *p-charts* for f_j do not signal, $j \neq i$ and $j=1,2,\ldots,K-1$. The Marcucci's method offers no help in interpreting which category has shifted when a signal occurs, so its diagnosis accuracy cannot be measured.

Also in this section, we compare the sensitivity of the *p-tree* method and the Marcucci's method to shifts in the category probabilities using ARL. The number of runs of each simulation is determined such that the standard error of every estimated ARLs, for both the *p-tree* and Marcucci methods, are less than 1.5% of the estimated ARLs. Diagnosis accuracy of the *p-tree* method is also measured as an average from all its signaling runs.

Now, we briefly present the Marcucci's control chart. The method uses the Pearson statistic:

$$X^2 = \sum_{i=1}^{K} \frac{(n_i - Np_i)^2}{Np_i} \tag{6}$$

where n_i is the realized number in category I, in a sample of size N, p_i is the known baseline fraction in category i. The X^2 statistic above is approximately Chi-square distributed assuming the process is in-control, the sample size N is greater than 167, and the expected occurrences Np_i are not too small. The upper control limit (UCL) of the Marcucci's chart is:

$$UCL = \chi^2_{(K-1,\alpha)}$$

where α is the false alarm rate and the upper tail area of the chi-square distribution with K-1 degrees of freedom. There is no lower control limit and the chart only indicates whether the process is in-control or not. If the chart tells that the process is out-of-control, we do not know whether it is too high or too low in any particular category.

4.1 *Diagnosis accuracy and ARLs for processes with three categories*

We conduct a simulation experiment to study the diagnosis accuracy of the *p-tree* method. The focus is on two scenarios: The first is called the Brick case where the baseline multinomial probabilities follow Marcucci's example. The second is called the Customer case where the baseline multinomial probabilities are more distributed. In this first simulation only processes with 3 categories, as in Figure 1, are considered. The experimental design is summarized in Table 3.

Factors	Levels
Baseline Probabilities	Brick p_1=0.95, p_2=0.03, and p_3=0.02 or f_1=0.95 and f_2=0.6, and Customer p_1=0.5, p_2=0.25, and p_3=0.25 or f_1=0.5 and f_2=0.5
ARL_0	20 and 200
Sample Size N	Brick case: 1,000 Customer case: 300
f_1	Brick case: from 0.95 to 0.945, 0.94, 0.935, and 0.93 Customer case: from 0.5 to 0.52, 0.54, 0.56, 0.58, and 0.6
f_2	Brick case: from 0.6 to 0.56, 0.52, 0.48, 0.44, and 0.4 Customer case: from 0.5 to 0.52, 0.54, 0.56, 0.58, and 0.6

Table 3. Factors and levels for examples with three categories

The ARL to a false alarm is denoted by ARL_0 and is equal to $1/\alpha$ since successive observations are independent. The levels for ARL_0 in the experiment are 20 and 200. This corresponds to α^* for the separate *p-chart*s equal to 0.0253 and 0.0025 respectively as in eqn. (3). Coleman et *al.* (2001) present control charts with ARL_0 of approximately 22 and 370 when monitoring business processes. Marcucci (1985) only uses an ARL_0 of 20 when monitoring samples sizes over 200 bricks.

Sample size parameters N are selected to guarantee a normal approximation to the Binomial distribution, the Chi-square approximation to the statistic X^2, as well as positive lower control limits, and upper control limits less than one for the individual *p-chart*s in the *p-tree* method. The shifts represent variations up to approximately three sigma of each tree fraction.

The independence property of the tree fractions $\hat{fi}$'s is confirmed by the Kendall nonparametric tests of independence (Kendall & Gibbons, 1990, p. 66) between $\hat{f}_1$ and $\hat{f}_2$ for both scenarios, for in-control data sets with 20,000 samples and a 0.1 level of significance.

Tables 4 and 5 show the diagnosis accuracy results of the *p-tree* method and the ARLs for the *p-tree* and Marcucci methods.

		ARL_0=20			ARL_0=200		
Tree Prob.	Tree Prob. Value	Diagnosis accuracy of *p-tree*	*p-tree* ARL	Marcucci ARL	Diagnosis accuracy of *p-tree*	*p-tree* ARL	Marcucci ARL
f_1	0.95		20.8	20.9		188.3	189.4
	0.95	0.76	10.3	9.0	0.88	46.6	42.4
	0.94	0.90	4.1	3.8	0.97	12.1	12.2
	0.94	0.95	2.1	2.0	0.99	4.3	4.5
	0.93	0.97	1.4	1.4	1.00	2.1	2.3
	0.60		20.7	20.6		188.3	189.3
f_2	0.56	0.69	12.8	12.6	0.74	77.0	76.8
	0.52	0.85	6.0	6.0	0.92	24.2	25.4
	0.48	0.92	3.0	3.1	0.98	8.7	9.6
	0.44	0.95	1.8	1.9	0.99	3.8	4.3
	0.40	0.97	1.3	1.4	0.99	2.1	2.4

Table 4. Diagnosis Accuracy (correct signals over the total signals) and ARL performances for Brick case, K=3

		ARL_0=20			ARL_0=200		
Tree Prob.	Tree Prob. Value	Diagnosis accuracy of *p-tree*	*p-tree* ARL	Marcucci ARL	Diagnosis accuracy of *p-tree*	*p-tree* ARL	Marcucci ARL
f_1	0.50		21.1	20.6		214.6	208.1
	0.52	0.71	11.7	11.9	0.80	87.0	94.1
	0.54	0.89	4.7	4.8	0.95	21.1	22.8
	0.56	0.94	2.2	2.3	0.98	6.0	6.6
	0.58	0.97	1.4	1.4	0.99	2.6	2.8
	0.60	0.97	1.1	1.1	1.00	1.5	1.6
f_2	0.50		21.1	20.6		214.6	208.1
	0.52	0.63	14.9	14.6	0.71	123.1	121.6
	0.54	0.81	8.0	7.9	0.90	45.7	45.1
	0.56	0.90	4.1	4.1	0.97	16.1	16.7
	0.58	0.94	2.4	2.5	0.99	6.9	7.3
	0.60	0.96	1.7	1.7	0.99	3.5	3.8

Table 5. Diagnosis Accuracy and ARL performances for Customer case, K=3

Tables 4 and 5 show that for both cases, the larger the shift from baseline, the better the diagnosis accuracy and the larger the ARL_0, the better the diagnosis accuracy for the same shift. For example, Table 5 shows that if f_1 shifts from 0.5 to 0.52 and ARL_0=20, the *p-tree* method signals correctly for 0.71 of the signals. If f_1 shifts from 0.5 to 0.6 and ARL_0=20, the *p-tree* method signals correctly 0.97 of the signals.

If the diagnosis accuracy measure is divided by the ARL, we get the fraction of correct signals over the total of samples. This fraction combines sensitivity with diagnosis accuracy and estimates the probability that the *p-tree* method signals correctly at any sample for a process that is out-of-control. For the Customer case in Table 5, if f_1 shifts to 0.52 and ARL_0=20, the fraction of correct signals over the total of samples is 0.06.

As expected, the *p-tree* diagnosis accuracy is better if the tree probability that shifts has a lower index, or comes first as a stage, because the stage's sample size is larger. For example, Table 5 shows that the diagnosis accuracy for shifts in f_1 is better than for shifts in f_2.

In terms of ARL comparisons, Table 4 shows that the Marcucci's method is slightly more sensitive than the *p-tree* method when monitoring shifts on f_1 in the Brick case, particularly when ARL_0=20, which is the case developed in Marcucci (1985). Table 5 shows that the *p-tree* method is slightly more sensitive than the Marcucci's method when monitoring shifts on f_1 in the Customer case. In general, Tables 4 and 5 show that the differences between the ARLs of both control charts are quite small. The significant contribution of the *p-tree* method is its value as a diagnosis tool.

Figure 3 shows a graph comparing the ARL performances when f_1 shifts in the Customer case. Although other graphs are not shown, this is a typical plot. In general, it is visually difficult to distinguish between the methods.

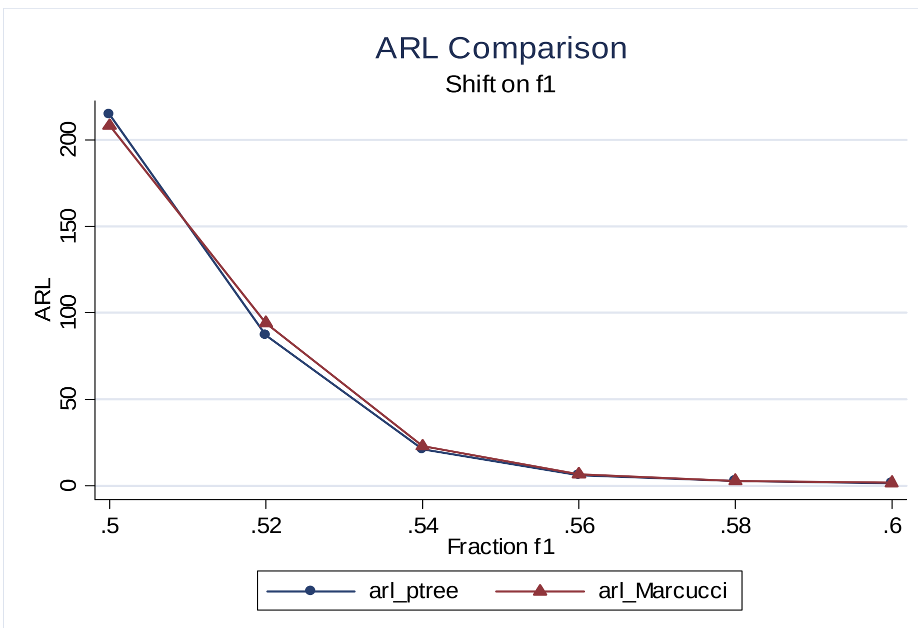

Figure 3. ARL comparison for shifts on f_1 in Customer case, K=3

4.2 *Diagnosis accuracy and ARLs for a process with six categories*

The simulation experiment in this section focuses on a Customer case with six categories, and it is summarized in Table 6. The tree baseline probabilities f_i are all equal to 0.5, i=1,2,…,5.

The sample size is N=1,000. The shifts represent variations up to approximately three sigma for each tree fraction.

Factors	Levels
Baseline Probabilities	p_1=0.5, p_2=0.25, p_3=0.125, p_4=0.0625, p_5=p_6=0.03125 or equivalently f_i=0.5 , i=1,…,5
ARL_0	20 and 200
Sample Size N	1,000
f_1	from 0.5 to 0.51, 0.52,…,0.55
f_2	from 0.5 to 0.51, 0.52,…,0.56
f_3	from 0.5 to 0.52, 0.54,…,0.60
f_4	from 0.5 to 0.52, 0.54,…,0.60
f_5	from 0.5 to 0.53, 0.56,…,0.68

Table 6. Factors and levels for simulation experiment with six categories

The α* for each p-control chart of $\hat{f_i}$, i=1, 2,.., 5 are obtained using eqn. (3) and are 0.0102 and 0.001 for ARL_0 of 20 and 200 respectively. The independence property among the tree fractions $\hat{f_i}$'s is confirmed by Kendall nonparametric tests of independence among the $\hat{f_i}$'s for an in-control data set with 20,000 samples at a 0.1 level of significance

Table 7 shows that the *p-tree*'s diagnosis accuracy is better if the shift size and/or ARL_0 are larger, and if the tree probability that shifts has a lower index. These results are similar to those in Tables 5 and 6 for three category case. The number of categories does not limit these advantageous features of the *p-tree* method.

		ARL$_0$=20			ARL$_0$=200		
Tree Prob.	Tree Prob. Value	Diagnosis accuracy of *p-tree*	*p-tree* ARL	Marcucci ARL	Diagnosis accuracy of *p-tree*	*p-tree* ARL	Marcucci ARL
f_1	0.50		20.0	19.9		204.7	192.3
	0.51	0.40	15.5	16.0	0.49	133.5	144.0
	0.52	0.71	7.6	8.4	0.83	43.4	54.4
	0.53	0.86	3.5	4.1	0.96	12.7	17.4
	0.54	0.92	2.0	2.3	0.98	4.6	6.4
	0.55	0.95	1.4	1.5	0.99	2.3	3.0
f_2	0.50		20.0	19.9		204.7	192.3
	0.51	0.31	17.9	18.8	0.38	156.4	178.7
	0.52	0.53	12.0	12.8	0.72	89.9	103.9
	0.53	0.73	7.0	7.7	0.87	34.0	43.9
	0.54	0.84	4.1	4.6	0.94	14.5	20.9
	0.55	0.90	2.5	3.0	0.97	6.8	10.1
	0.56	0.93	1.8	2.0	0.98	3.7	5.3
f_3	0.50		20.0	19.9		204.7	192.3
	0.52	0.39	15.5	16.7	0.50	127.9	145.8
	0.54	0.70	7.7	8.5	0.85	41.4	52.6
	0.56	0.86	3.6	4.2	0.95	12.1	17.3
	0.58	0.92	2.0	2.3	0.98	4.5	6.5
	0.60	0.95	1.4	1.5	0.99	2.2	3.0
f_4	0.50		20.0	19.9		204.7	192.3
	0.52	0.30	18.0	18.1	0.36	169.9	162.5
	0.54	0.54	12.0	12.4	0.67	90.3	89.8
	0.56	0.73	7.0	7.6	0.86	36.3	42.2
	0.58	0.84	4.0	4.5	0.94	15.1	19.8
	0.60	0.90	2.5	2.9	0.97	7.0	9.6
f_5	0.50		20.0	19.9		204.7	192.3
	0.53	0.31	17.7	17.5	0.37	158.6	140.5
	0.56	0.55	11.5	11.0	0.68	79.1	69.2
	0.59	0.75	6.5	6.4	0.88	31.0	31.0
	0.62	0.85	3.6	3.8	0.95	12.2	14.1
	0.65	0.91	2.2	2.5	0.98	5.6	7.2
	0.68	0.93	1.6	1.8	0.99	3.0	4.0

Table 7. Diagnosis Accuracy and ARL performances for Customer case, K=6.

Table 7 also shows for ARL comparisons, the *p-tree* method is slightly better than the Marcucci's method when monitoring shifts on f_1, f_2 or f_3. However, the Marcucci's method is slightly more sensitive when monitoring small shifts on f_4 and f_5, particularly when ARL$_0$=200. Notice that the smaller the index of the fraction monitored, the larger the stage's sample size in the *p-tree* method, and the better the ARL performance of the *p-tree* method over the Marcucci's method.

Figure 4 shows a graph comparing the ARL performances when f_4 shifts, which is a typical plot that shows the closeness between the ARLs of both methods.

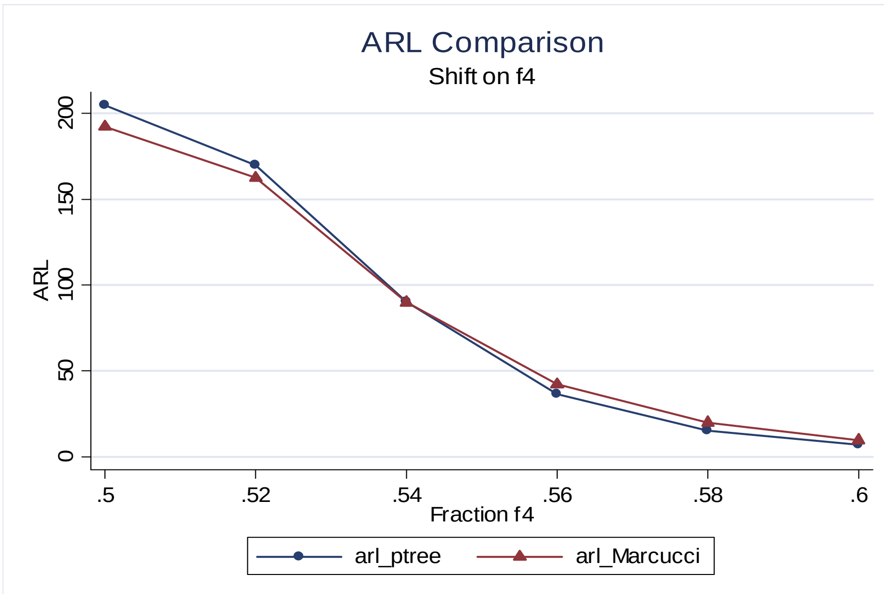

Figure 4. ARL comparison for shifts on f_4, K=6 (shows typical similarity between methods)

5 Conclusion

Transaction processes in the service industry or manufacturing processes with multiple categories can be sampled and modeled as multinomial processes. We propose to monitor any multinomial process by decomposition into independent binary stages using a probability tree.

If a conditional probability related to a tree's stage shifts, the *p-tree* method accurately detects it. As shown on the results, the larger the shift size, or the larger the ARL_0, or the lower the index of the tree's stage that shifts, the better the diagnosis accuracy of the *p-tree*'s signal. By contrast, a signal in the Marcucci's method may not be interpreted or measured, because the counts (n_i) across the categories are negatively correlated among them.

We showed also the limitations of other recent methods (Bayesian's based) for monitoring multinomial processes. These methods can not provide interpretations for customer transactions, because they can not take into account the correlation among categories.

Comparisons of ARL results of the *p-tree* and Marcucci's methods with three and six categories show that the *p-tree* method has similar sensitivity (or even slightly better) than the Marcucci's method. The *p-tree* method has the flexibility to fit any multinomial process, as long as the fractions in the converted tree are approximately normally distributed. Future research issues are: nonnormal tree fractions, problems with a non obvious order of categories in the tree, and problems with simultaneous shifts on several fractions. Processes with multiple stages, where every stage has many categories could be fully investigated because those models could serve to monitor complex and large interaction processes between customers and organizations either in the private or public sectors.

References

Coleman, S.Y., Arunakumar, G., Foldvary, F. and Feltham, R. (2001) SPC as a tool for creating a successful business measurement framework. *Journal of Applied Statistics*, **28**(3 & 4), 325-334.

Johnson, N.L., Kotz, S. and Balakrishnan, N. (1996) *Discrete Multivariate Distribution,* Wiley Series in Probability and Statistics, John Wiley & Sons, Inc.

Kemp, C.D. and Kemp, A.W. (1987) Rapid generation of frequency tables. *Applied Statistics*, **36**(3), 277-282.

Kendall, M.G. and Gibbons, J.D. (1990) *Rank Correlation Methods,* 5th Edition, New York, Oxford University Press.

Laviolette, M. (1995) Bayesian monitoring of multinomial processes. *The Journal of the Industrial Mathematics Society*, **45**, 41-49.

Lowry, C.A., Woodall, W.H., Champ, C.W. and Rigdon, S.E. (1992) A multivariate exponentially weighted moving average control chart. *Technometrics*, **34**(1), 46-53.

Marcucci, M. (1985) Monitoring multinomial processes. *Journal of Quality Technology*, **17**(2), 86-91.

Montgomery, D.C. (2005) *Introduction to Statistical Quality Control*, 5th Edition, John Wiley & Sons, Inc., New York, NY.

Montgomery, D.C. and Runger, G.C. (2002) *Applied Statistics and Probability for Engineers*, 3rd Edition, John Wiley & Sons, Inc., New York, NY.

Property Tax Assessment Study Commission (1986) *Report of the Property Tax Assessment Study Commission, State of New Jersey*, The Commission.

Rafool, M. (2002) *A Guide To Property Taxes: An Overview*, NCSL Fiscal Affairs Program, National Conference of State Legislatures (www.ncsl.org).

Shiau, J.H., Chen, C. and Feltz, C.J. (2005) An empirical Bayes process monitoring technique for polytomous data. *Quality and Reliability Engineering International*, **21**, 13–28.

Simonoff, J.S. (2003) *Analyzing Categorical Data*, Springer texts in statistics, Springer-Verlag New York, Inc.

Skinner, K.R., Runger, G.C. and Montgomery, D.C. (2006) Process monitoring for multiple count data using a deleted-Y statistic. *Quality Technology and Quantitative Management*, **3**, 247–262.

Tucker, G.R., Woodall, W.H. and Tsui, K-L. (2002) A control chart for ordinal data. *American Journal of Mathematical and Management Sciences*, **22**(1), 31-48.