A COMPARATIVE STUDY OF BOOK SELLING RECOMMENDATION TECHNIQUES

PATTHAMA SAETING

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF SCIENCE
(TECHNOLOGY OF INFORMATION SYSTEM MANAGEMENT)
FACULTY OF GRADUATE STUDIES
MAHIDOL UNIVERSITY
2013

Thesis
entitled
A COMPARATIVE STUDY OF BOOK SELLING RECOMMENDATION TECHNIQUES

…………………………………………………..
Miss. Patthama Saeting
Candidate

………………………………………………
Lect. Mingmanas Sivaraksa,
Ph.D. (Neural Computing Research Group)
Major advisor

………………………………………………
Lect. Tanasanee Phienthrakul,
Ph.D. (Computer Engineering)
Co- advisor

………………………………………………
Lect. Lalita Narapiyakul,
Ph.D. (Computer Science)
Co-advisor

………………………………………………
Prof. Banchong Mahaisavariya,
M.D., Dip. (Thai Board of Orthopedics)
Dean
Faculty of Graduate Studies,
Mahidol University

………………………………………………
Lect. Supaporn Kiattisin,
Ph.D.
(Electrical and Computer Engineering)
Program Director
Master of Science Program in
Technology of Information System
Management
Faculty of Engineering,
Mahidol University

Thesis
entitled
A COMPARATIVE STUDY OF BOOK SELLING RECOMMENDATION TECHNIQUES

was submitted to the Faculty of Graduate Studies, Mahidol University
for the degree of Master of Science
(Technology of Information System Management)
on
November 22, 2013

……………………………………………
Miss. Patthama Saeting
Candidate

……………………………………………
Asst. Prof. Bunlur Emaruchi,
Ph.D. (Environmental Systems Engineering)
Chair

……………………………………………
Lect. Mingmanas Sivaraksa,
Ph.D. (Neural Computing Research Group)
Member

……………………………………………
Lect. Tanasanee Phienthrakul,
Ph.D. (Computer Engineering)
Member

……………………………………………
Asst. Prof. Worasit Choochaiwattana,
Ph.D. (Information Science)
Member

……………………………………………
Lect. Lalita Narapiyakul,
Ph.D. (Computer Science)
Member

……………………………………………
Prof. Banchong Mahaisavariya,
M.D., Dip. (Thai Board of Orthopedics)
Dean
Faculty of Graduate Studies
Mahidol University

……………………………………………
Lect. Worawit Israngkul,
M.S. (Technical Management)
Dean
Faculty of Engineering
Mahidol University

ACKNOWLEDGEMENTS

This thesis was brought to a successful completion by the great kindness of my advisor, Lect. Mingmanas Sivaraksa, my co-advisers, Lect. Tanasanee Phienthrakul and Lect. Lalita Narapiyakul, who have offered timely assistance, advice and revision in the preparation of this thesis dissertation for which kindness I am deeply grateful. I would like to take this opportunity to express my sincere appreciation to all three of the abovementioned advisors.

I would also like to thank the chairman of the thesis board of examiners, Lect. Supaporn Kiattisin, and thesis board examiner, Asst. Prof. Worasit Choochaiwattana, for offering helpful ideas and suggestions together with kindness in correcting research errors.

Much gratitude is also due to the faculty of professors and personnel in the Technology Information System Management Program, Faculty of Engineering, Mahidol University for sharing knowledge, offering assistance and providing support in so many ways.

I would also like to express my heartfelt gratitude to Kledthai Co., Ltd. for the data used in the case study for this research

I would to express many thanks to all of my friends for motivating encouragement throughout the course of the research. I am also thankful to the junior students in the Faculty of Computer Engineering for their concerned inquiries about the completion of the research. I am truly grateful for the motivation and encouragement received from these extraordinary friendships.

In closing, I would like to thank the most important people in my life, namely, my family to whom I will forever be indebted for their endless support, concern, love and understanding throughout the course of the research.

Patthama Saeting

A COMPARATIVE STUDY OF BOOK SELLING RECOMMENDATION TECHNIQUES

PATTHAMA SAETING 5236785 EGTI/M

M.Sc. (TECHNOLOGY OF INFORMATION SYSTEM MANAGEMENT)

THESIS ADVISORY COMMITTEE: MINGMANAS SIVARAKSA, Ph.D., TANASANEE PHIENTHRAKUL, Ph.D., LALITA NARAPIYAKUL, Ph.D.

ABSTRACT

In today's highly competitive world, quick decisions should be made without major mistakes. The decisions based solely on individuals are frequently complicated and are also more time-consuming in order to obtain the optimal result. These possibly cause errors as a result of several negative effects on businesses. Product distribution is a key problem in logistic system, such as how to manage inventory to satisfy customers need without affecting product quality. Therefore, it is essential to have a plan for efficient product distribution, and how to control the quantity of the product. Product recommendation is another important problem when introducing a new products line that has arrived. Manufactures and retailers do not know which customers would be interested in the new products; but if we distribute product evenly to every customer, and wait for responses, that will waste time in the distribution of the products.

This research proposes three methods that have standard or are used in Recommender System; Collaborative Filtering, Artificial Neural Network, and Decision Tree., for solving problems of book distribution and book recommendations for customers who would like to be bookstore owners by using a case study of book distribution from Kledthai Book Distribution Company. Input data was divided into 2 different types which were comprised of numeric data and nominal data. The result will be displayed in the form of prediction that will help monitor the result and determine errors in predicting quantity of sales. According to the experiment, a decision tree method was shown to be the most accurate of both data types, which has an accuracy at 40.10% of input data that is numeric data and 40.58% of input data that is nominal data. This researcher concludes that the decision tree method has the highest accuracy value, or was able to yield the most accurate prediction.

KEY WORDS: RECOMMENDER SYSTEM / COLLABORATIVE FILTERING / ARTIFICIAL NEURAL NETWORK / DECISION TREE

126 pages

การศึกษาเปรียบเทียบเทคนิคการแนะนำการขายหนังสือ

A COMPARATIVE STUDY OF BOOK SELLING RECOMMENDATION TECHNIQUES

ปัทมา แซ่ติ้ง 5236785 EGTI/M

วท.ม. (เทคโนโลยีการจัดการระบบสารสนเทศ)

คณะกรรมการที่ปรึกษาวิทยานิพนธ์: มิ่งมานัส ศิวรักษ์, Ph.D., ธนัสนี เพียรตระกูล, Ph.D., ลลิตา นฤปิยะกุล, Ph.D.

บทคัดย่อ

เนื่องจากปัจจุบันมีการแข่งขันกันสูง การดำเนินการบางอย่างก็ต้องใช้การตัดสินใจที่รวดเร็ว และไม่มีข้อผิดพลาดหรือมีข้อผิดพลาดที่น้อยที่สุดที่จะรับได้ ซึ่งบางเรื่องการตัดสินใจโดยคนก็เป็นเรื่องยาก ต้องมีการวิเคราะห์ คำนวณที่ซับซ้อนวุ่นวายต้องใช้เวลานานกว่าจะได้ผลที่ต้องการ อาจทำให้เกิดข้อผิดพลาดที่ส่งผลกระทบต่อธุรกิจ ปัญหาการกระจายสินค้าเป็นหนึ่งปัญหาที่มีความสำคัญ เช่นจะจัดการสินค้าคงคลังอย่างไรให้ลูกค้าพึงพอใจโดยไม่ให้มีผลกระทบต่อคุณภาพจึงต้องมีการวางแผนการกระจายสินค้า และปริมาณที่เหมาะสมต่อการกระจายสินค้าไปยังลูกค้าแต่ละราย ปัญหาในเรื่องของการแนะนำสินค้าเป็นอีกหนึ่งปัญหาที่สำคัญเนื่องจากเมื่อมีสินค้าใหม่ เราไม่สามารถทราบได้ว่าลูกค้ารายใดที่จะให้ความสนใจต่อสินค้าใหม่นี้ ถ้ากระจายสินค้าไปยังลูกค้าทุกรายเพื่อรอการตอบรับกลับมาจะเป็นการสิ้นเปลืองเวลาในการขนส่งสินค้า

งานวิจัยฉบับนี้ได้นำเสนอ 3 วิธีที่มีมาตรฐานหรือนิยมนำมากับระบบให้คำแนะนำ ซึ่งได้แก่ ตัวกรองเชิงร่วมมือ โครงข่ายประสาทเทียม และ ต้นไม้ตัดสินใจ เพื่อนำมาแก้ปัญหาการกระจายหนังสือ และการแนะนำหนังสือของร้านหนังสือที่เปรียบเสมือนร้านค้าปลีก โดยใช้ข้อมูลการกระจายหนังสือของบริษัทเคล็ดไทยเป็นกรณีศึกษา โดยข้อมูลนำเข้าจะแบ่งเป็น 2 ชนิด คือข้อมูลตัวเลขและข้อมูลนามบัญญัติ ผลลัพธ์ของแต่ละวิธีจะแสดงในรูปแบบของการคาดการณ์ซึ่งจะช่วยแสดงผลลัพธ์ที่ชัดเจน และค่าความผิดพลาดในการคาดการณ์ จากการทดลองวิธีต้นไม้ตัดสินใจให้ค่าความถูกต้องมากที่สุดในทั้ง 2 ชนิดข้อมูลนำเข้า คือร้อยละ 40.10 ของข้อมูลตัวเลข และ ร้อยละ 40.58 ของข้อมูลนามบัญญัติ ซึ่งสรุปได้ว่าวิธีต้นไม้ตัดสินใจมีค่าความถูกต้องสูงที่สุดหรือสามารถคาดการณ์ผลลัพธ์ที่มีความถูกต้องมากที่สุด

126 หน้า

CONTENTS

CONTENTS (cont.)

CONTENTS (cont.)

LIST OF TABLES

LIST OF TABLES (cont.)

Table **Page**

LIST OF FIGURES

CHAPTER I
INTRODUCTION

1.1 Background and Problem statements

In today's competitive world, quick decisions making should be made no or least error should be found in those decisions. Decisions based purely on human are unreliable because they must be involved in complicated computing and take a long time before the optimal result is found. These may cause errors that affect businesses. Product differentiation and reliable services with rapid response are key factors to satisfy to customers.

Product distribution is a key problem in logistic system, such as how to manage inventory to satisfy customers need without affecting product quality. Therefore, it is essentially to have a plan for effective product distribution and to control a quantity of product for each customer given limited number of products. As a result, it will reduce the waste shipping time but it must determine the amount of spare products in the inventory for emergency case, too. However if these problems are not solved, they will incur a cost of products.

Product recommendation is another important problem because when new products arrived, we do not know which customers would be interested in the particular products but if we distribute product evenly to every customer and wait for responses that will waste more time in distributing the products. For example, if there are a few customers who interested in a new arrival product but we send this product to every customer, therefore a cost of product will be very high. In another case, we randomly distribute a product to some customers; in the positive way either every customer interested in the product or only a half of the customers interested in sent product. The worst is that no customers interested in the sent product. Therefore, it is difficult to randomly selected customers who are interested in the product, so it contains a risk in business.

As a result, there should be some tools for analyzing and estimating product distribution to each customer or determining in distributing new arrival products to appropriate customers, include determining quantity of products in stock for the best beneficial. These will help to reduce costs and risks of overstocking. Eventually customers will get the most appropriate quantity of products. Furthermore, calculating the best quantity of products for distributing to customers by using historical order in term of product type and quantity which have little or limited information will help rapidly, conveniently and efficiency distribute products and help to scope a group of new attractive products to be recommended in the future which do not require high expertise.

According to the mentioned problems, we decided to study, analyze, and compare product recommendation systems for solving the problem of product distribution and product recommendation by using various methods which is appropriate for the different volume of information and errors.

Purpose of this research is to solve a problem of book distribution and book recommendation for customers which like to be a bookstore. By using a case of Kledthai book distribution company. This research compares “Collaborative Filtering”, “Artificial Neural Network”, and “Decision Tree” then displays a result of each method in form of prediction which will help to monitor the result clearly.

1.2 Objective of study

1. To study customers behavior by determining previous purchasing history.
2. To study and analyze Artificial Intelligence methods; for recommending products to customers and assess a quantity of product for the most appropriate distribution.
3. To compare each method of Artificial Intelligence from the previous learning which will help to see a difference from each result for easy deciding to choose an appropriate method.

1.3 Scope of work

1. The book distribution data during year 2009 to 2011, consist of
    a. Book of 5000 Books are divided into 21 categories
    b. Publisher of 216 publishers.
    c. Store of 10 stores
2. Three methods are selected that have standard or used in recommender system.
3. The results will be in the form of comparisons between 3 selected methods.

1.4 Expected Results

1. To know the most appropriate method which can be used to recommend the best product to offer to customers of the book industry.
2. To know the most important attributes which influence the prediction.
3. To be able to efficiency and effectively allocate appropriate products to customers.
4. Accuracy measure is the key method for comparing different techniques and to measure the performance of the prediction.
5. This research can be guidance for product recommending system or to learn about the recommending product system.

CHAPTER II
LITERATURE REVIEW

2.1 Recommender Systems

Recommender Systems are software tools and techniques providing suggestions to make items useful for users. The suggestions provided are aimed at supporting users in various decision-making processes, such as what items to buy, what music to listen to, or what news to read. Recommender systems have proven to be valuable means for online users to cope with information overload and have become one of the most powerful and popular tools in electronic commerce (e-commerce). Correspondingly, various techniques for recommendation generation have been proposed. During the last decade, many of have also been successfully deployed in commercial environments [1]. Recommendation algorithms are best known for their use on e-commerce web sites, where they use the input of customer interests to generate a list of recommended items. Many applications use only the items customers purchase and explicitly rate to represent their interests, but they can also use other attributes, including items viewed, demographic data, subject interests and favorite artists [2]. Recommender systems play an important role, particularly in an e-commerce environment, as a new marketing strategy. Although an assortment of recommendation techniques has been developed recently, collaborative filtering (CF) has been known to be the most successful recommendation technique [3].

Recommendation systems are rapidly advancing on the Web as an effective way to connect consumers with products and services they would most likely purchase based on past actions and interests. With recommendation technology, web properties can become even more dynamic as they deliver direct matching links based on the aggregation of user patterns. When building a user's profile, a distinction is made between explicit and implicit forms of data collection by analyzing the user's social network and discovering similar likes and dislikes. The goal of a recommender system is to generate meaningful recommendations to a collection of users for items or products of potential interest. Suggestions for books on Amazon, or movies on Netflix, are real world examples of the operation of industry-strength recommender systems. The design of such recommendation engines depends on the domain and the particular characteristics of the data available. For example, movie watchers on Netflix frequently provide ratings on a scale of 1 (disliked) to 5 (liked). Such a data source records the quality of interactions between users and items. Additionally, the system may have access to user-specific and item-specific profile attributes such as demographics and product descriptions, respectively. Recommender systems differ in the way they analyze these data sources to develop notions of affinity between users and items which can be used to identify well-matched pairs [4].

2.2 Collaborative Filtering (CF)

The famous electronic commerce websites, Amazon and CD-Now, have employed a CF technique to recommend products to customers and it has improved the quality and efficiency of their services. The CF assumes that a good way to find a certain user’s interesting content is to find other people who have similar interests with him. Typically, a recommender system compares the user's profile to some reference characteristics. These characteristics may be from the information of each item (the content-based approach) or the user's social environment (the collaborative filtering approach) [5].

Collaborative filtering (CF) systems work by collecting user feedback in the form of ratings for items in a given domain and exploiting similarities in rating behavior amongst several users in determining how to recommend an item. The CF method can be further subdivided into neighborhood-based and model-based approaches. Neighborhood-based methods are also commonly referred to as memory-based approaches [6, 7, 8].

The system can make predictions or recommendations based on the correlation coefficients. The current popular algorithm to compute correlation coefficients is Pearson r Correlation Coefficient. Given two user’s list of ratings as $x = [x_1, \dots, x_t]^T$ and $y = [y_1, \dots, y_t]^T$, the standard Pearson r Correlation Coefficient is used to measure the similarity (s_r) between two lists of ratings and calculated as follows [9] :

$$s_r = \frac{\sum_{i=1}^{t}(x_i - x_{avg})(y_i - y_{avg})}{\sqrt{\left(\sum_{i=1}^{t}(x_i - x_{avg})^2\right)\left(\sum_{i=1}^{t}(y_i - y_{avg})^2\right)}} \tag{2.1}$$

Predictions are generally computed as the weighted average of deviations from the neighbor's mean as in the following:

$$P_{a,i} = \bar{r}_a + \frac{\sum_{u \in K}(r_{u,i} - \bar{r}_u) \times w_{a,u}}{\sum_{u \in K} w_{a,u}} \tag{2.2}$$

where $P_{a,i}$ is the prediction for the active user a for item i, K is the neighborhood or set of most similar users. $w_{a,u}$ is a measure of similarity between the neighbors u and the active user a. The most commonly used measure of similarity is the Pearson correlation coefficient between the ratings of the two users below:

$$w_{a,u} = \frac{\sum_{i \in I}\left((r_{a,i} - \bar{r}_a)\right)(r_{u,i} - \bar{r}_u)}{\sigma_a \times \sigma_u} \tag{2.3}$$

where I is the set of items rated by both users, $r_{u,i}$ is the rating given to item i by user u, and $\bar{r}_u$ is the mean rating given by user u, σ_a and σ_u are variances of user a and u' s rating values, respectively [10,11].

$$\sigma_a = \sqrt{\frac{\sum_{i=1}^{m}(r_{a,i} - \bar{r}_a)^2}{m}} \tag{2.4}$$

where $r_{a,i}$ is the rating given to item i by user a, and $\bar{r}_a$ is the mean rating given by user a, m is the set of co-rated items [12].

2.3 Artificial Neural Network (ANN)

An artificial neural network is a system based on the operation of biological neural networks, in other words, is a simulation of the biological neural system. Although modern computing is truly advanced, there are certain tasks a program made for a common microprocessor is unable to perform. Basically, an artificial neural network is a system. A system is a structure that receives input, processes the data and provides output.

Error information is fed back to the system which makes all adjustments to the parameters in a systematic fashion (commonly known as the learning rule). This process is repeated until the desired output is acceptable. It is important to notice that the performance hinges heavily on the data.

2.3.1 Training an artificial neural network

A neural network has to be configured in such a way that the application of a set of inputs produces (either 'direct' or via a relaxation process) a desired set of outputs. Various methods to set the strengths of the connections exist. One way is to set the weights explicitly by using a priori knowledge. Another way is to 'train' the neural network by supplying it with teaching patterns and letting it change its weights according to a certain learning rule.

We can categorize learning situations into the following three distinct sorts:

- **Supervised Learning** or Associative Learning occurs when the network is trained by providing input and matching output patterns. These input-output pairs can be provided by an external teacher, or by the system which contains the neural network (self-supervised). Figure 2.1, for example, shows the structure of supervised learning.

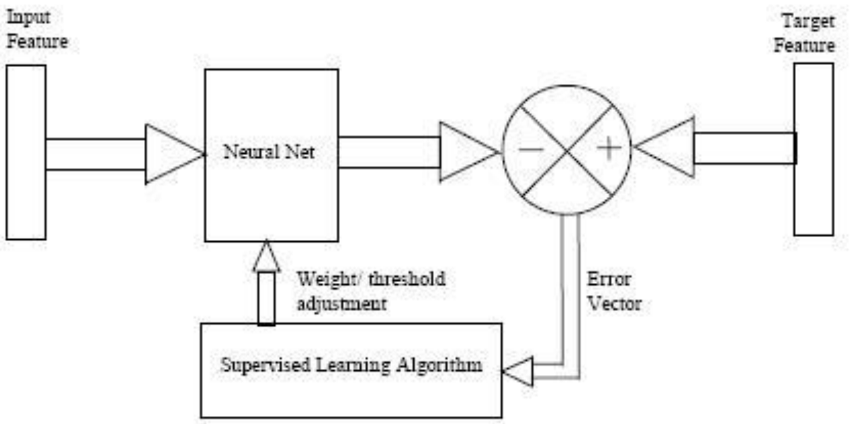


Figure 2.1 Supervised learning structures [13].

- **Unsupervised learning** or Self-Organisation occurs when an (output) unit is trained to respond to clusters of patterns within the input. In this paradigm, the system is supposed to discover statistically salient features of the input population. Unlike the supervised learning paradigm, there is no a priori set of categories into which the patterns are to be classified; rather, the system must develop its own representation of the input stimuli.

- **Reinforcement Learning** may be considered as an intermediate form of the above two types of learning. Here, the learning machine performs a certain action on the environment and receives a feedback response from the environment. The learning system grades its action as either good (rewarding) or bad (punishable) based on the environmental response and adjusts its parameters accordingly. Generally, parameter adjustment is continued until an equilibrium state occurs, following which no more changes will be made in its parameters. The self-organizing neural learning may be categorized under this type of learning [13].

2.3.2 Feed-Forward Networks

The basic architecture consists of three types of neuron layers: input, hidden and output layers, as show in Figure 2.2

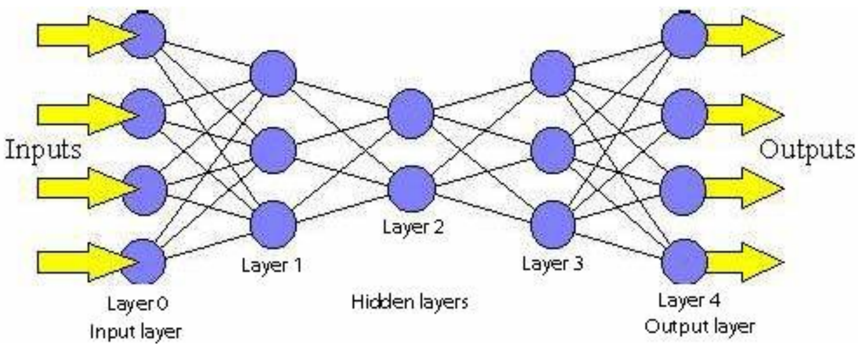


Figure 2.2 Feed-forward networks [14].

Feed-forward networks have the following characteristics [14]**:**

1. Perceptrons are arranged in layers, with the first layer taking in inputs and the last layer producing outputs. The middle layers have no

connection with the external world. Hence, they are called hidden layers.

2. Each perceptron in a layer is connected to every perceptron on the next layer. Thus, information is constantly "fed forward" from one layer to the next and this explains why these networks are called feed-forward networks.
3. There is no connection among perceptrons in the same layer.

2.3.3 Multi - Layer Perceptron (MLP)

The multilayer perceptron consists of a system of simple interconnected neurons, or nodes, as illustrated in Figure 2.3 which is a model representing a nonlinear mapping between an input vector and an output vector. The nodes are connected by weights and output signals which are a function of the sum of the inputs to the node modified by a simple nonlinear transfer, or activation, function. It is the superposition of many simple nonlinear transfer functions that enables the multilayer perceptron to approximate extremely non-linear functions. If the transfer function is linear then the multilayer perceptron will only be able to model linear functions. The output of a node is scaled by the connecting weight and fed forward to be an input to the nodes in the next layer of the network. This implies a direction of information processing; hence, the multilayer perceptron is known as a feed-forward neural network. The architecture of a multilayer perceptron is variable but generally consists of several layers of neurons. The input layer plays no computational role but merely serves to pass the input vector to the network. The terms input and output vectors refer to the inputs and outputs of the multilayer perceptron and can be represented as single vectors. A multilayer perceptron may have one or more hidden layers and finally an output layer. Multilayer perceptron are described as being fully connected, with each node connected to every node in the next and previous layer [15].

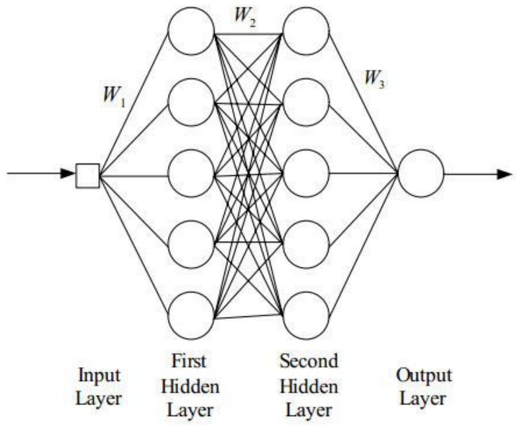


Figure 2.3 Multilayer perceptron with two hidden layers [16].

2.3.4 Training a multilayer perceptron – The back propagation algorithm

Learning in feed-forward networks belongs to the realm of supervised learning, in which pairs of input and output values are fed into the network for many cycles, so that the network 'learns' the relationship between the input and output [17]. For example in Figure 2.4 shows the method of training multilayer perceptrons by using a back propagation algorithm.

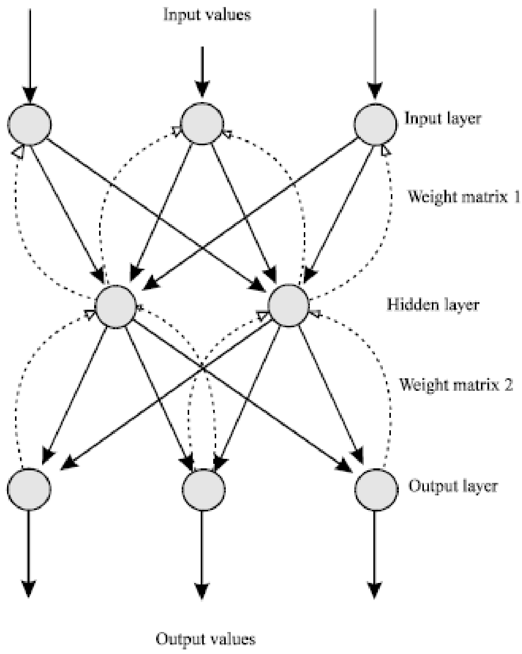


Figure 2.4 Multilayer perceptron training procedure using a back propagation algorithm [18].

The solid arrows represent the forward operation, while the dotted arrows represent the backward operation. The circles represent the neurons. Each connection between the layers has a specific weight that is stored in weight matrix. The final output obtained from the summation of the inputs in the final stage for each neuron located in the output layer [19].

The equation used training multilayer perceptron is as follows:

1. The Summation Function is

$$v_k = \sum_{j=1}^{p} w_{kj} x_j \tag{2.5}$$

where x_j is input data, w_{kj} is the weight of input data, p is the number of input data.

2. The Activation Function is

$$y_k = f\left(\sum_{j=1}^{p} w_{kj} x_j\right) \tag{2.6}$$

where f is the activation function, this research uses the sigmoid function because the classification problem-involving attributes we often use sigmoid function as in the following example:

$$y_j = \frac{1}{1+exp(-a_j)} \tag{2.7}$$

$-a_j$ are then transformed by the activation function of the output layer to give output values y_j .

3. The error of the actual value is calculated and the target value is given as follows:

$$\varepsilon^2 = \sum_k \left(o_k - \hat{o}_k\right)^2 \tag{2.8}$$

where o_k is the target value, $\hat{o}_k$ is the target output value.

4. Each round of learning will have the weights adjusted as follows:

$$\Delta w_{kj} = -\alpha \cdot \frac{\partial \varepsilon^2}{\partial w_{kj}} \tag{2.9}$$

$$w_{kj(new)} = w_{kj(old)} + \Delta w_{kj} \tag{2.10}$$

where α is the learning rate [20].

2.4 Decision Tree

Whether Data mining is used for a business or other purposes, it requires a decision tree to help in the decision-making and usually consists of rules in the form "if condition then result" such as "IF Income = High and Married = No THEN Risk = Poor". According to its nature, the decision tree is similar to an upside-down tree. The first node is the root node where each node represents an attribute, each branch shows the test results and the leaf node shows a class has been defined. Figure 2.5 shows an example of a decision tree.

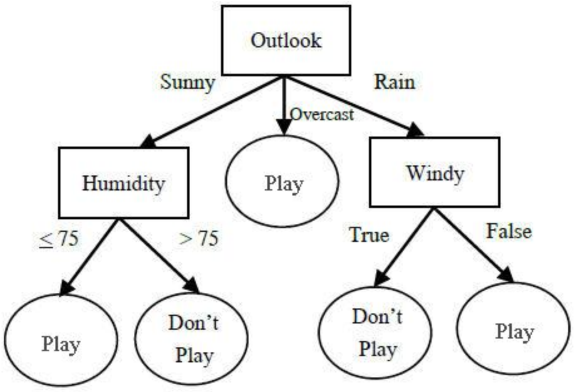


Figure 2.5 Example of Decision Tree [21].

ID3 is a method used to divide the group in order to create the best tree. The concept of ID3 involves trying to find the best separators in the data classification (Classifier) by using the information theory where the measurement will be used in the decision making and variables will be used in the prediction or divided by the type of data. Sample are data used to learn (Training Sample) with a variable target (Target Attribute) [22] .The steps are as follows:

1. Test all of the samples received; determine whether or not the Target Attribute value is positive. If so, return the positive node and exit. If not, go to the next step.
2. Test all of the samples received, determine whether or not the Target Attribute value is negative. If so, return the negative node and exit. If not, go to the next step.
3. Test the other attributes which are not the Target Attribute to

determine whether or not they have a value. If they have no values, finish the task. If they have values, the procedure is as follows:

3.1 Find the entropy by

$$Info(D) = -\sum_{i=1}^{m} p_i log(p_i) \tag{2.11}$$

where $Info(D)$ is the measurement of the information.

D represents a set of data.

p represents the probability of the Class.

3.2 Find information by

$$Info_A(D) = -\sum_{j=1}^{V} \frac{|D_j|}{|D|} \times I(D_j) \tag{2.12}$$

where A represents a variable or attribute.

D represents a set of data.

3.3 Find the Information Gain of all attributes that are not the target attributes, which attributes have the most gain value and are defined as the top node (Root Node) by

$$Gain(A) = Info(D) - Info_A(D) \tag{2.13}$$

where $Gain(A)$ is the reliability value.

3.4 Repeat (Loop) for all possible values of attributes, A, by the following procedure:

1. Select a sample of Attribute A with the greatest value.
2. Repeat from (2.12) until a complete decision tree has been created by using each parameter that is a value of the all samples selected, the target value and the value of the attribute that deletes attribute A.

The C4.5 classification algorithm which represents the results in the form of a decision tree is the method for creating trees as in ID3, but includes the process of updating the tree for better similarity. The J48 algorithm is the improved version of the C4.5 algorithm or can be called an optimized implementation of the C4.5. The output of J48 is the decision tree. The decision tree is similar to the tree structure with the roots as nodes, namely intermediate nodes and leaf nodes. Each node in the tree consists of a decision and that decision leads to a result. A decision tree divides the input space of a data set into mutually exclusive areas where each area has a label, a value or an action to describe its data points. Splitting criterion is used to calculate which attribute is the best to split that portion tree of the training data reaching a particular node [21, 23].

2.5 K-Fold Cross-Validation

In the k-fold cross-validation technique, the data is first partitioned into k equal (or nearly equal) segments or folds. Subsequently, k iterations of training and validation are performed within each iteration, a different fold of the data is held-out for validation and the remaining k - 1 folds are used for learning. Data is commonly stratified prior to being split into k folds. Stratification is the process of rearranging the data to ensure each fold is a good representative of the whole [24]. For example, in Figure 2.6 shows input data partitioned by using 5-fold cross validation.

Iteration 1: train on	2	3	4	5	, test on	1
Iteration 2: train on	1	3	4	5	, test on	2
Iteration 3: train on	1	2	4	5	, test on	3
Iteration 4: train on	1	2	3	5	, test on	4
Iteration 5: train on	1	2	3	4	, test on	5

Figure 2.6 Example of 5 - fold cross validation [25].

2.6 Literature Review

Liu Hongsheng, Yan Yongcai (2010) [26] proposed that the critical marketing environment is the restraint conditions of enterprise business activities, which has extremely important influence on the existence and development of enterprises. Meanwhile, the critical marketing environment is also one of the key factors influencing consumer decision-making. On the basis of studying the factors influencing consumer decision-making and the macro marketing environment of enterprises, the above researchers explored how the critical marketing environment influences consumer decision-making which found the critical marketing environment to be a micro environment consisting of manufacturers, product prices, product properties and product quality.

Maria Cleofe´ Valverde Ramı´rez, et. al. (2005) [27] researched forecasting rainfall, which uses an artificial neural network (ANN) technique to construct a nonlinear mapping between output data from a regional ETA model run at the Center for Weather Forecasts and Climate Studies/National Institute for Space Research/Brazil to determine the surface rainfall data for the region of Sao Paulo State, Brazil. The objective was to generate site-specific quantitative forecasts on daily rainfall. The test was performed at six locations in Sao Paulo State during the austral summer and winter periods from 1997 to 2002. The analysis was made using a feed forward neural network and a resilient propagation learning algorithm. The meteorological variables from the ETA model (potential temperature, vertical component of the wind, specific humidity, air temperature, precipitable water, relative vorticity and moisture divergence flux) were used as input data for the trained networks generating rainfall forecasts. Additionally, predictions with a multiple linear regression model (MLR) were compared to those of ANN. In order to evaluate the rainfall forecast skills over the studied region, statistical analysis was performed. The results show the ANN forecasts to be superior to the forecasts obtained by the linear regression model, thus revealing great potential for an operational suite. Therefore, ANN is better than multiple linear regression because it can be used to construct nonlinear mapping and it is more accurate in rainfall forecasting.

S.J. Park, C.S. Hwang, P.L.G. Vlek (2005) [28] investigated the potential of predicting crop yield responses under varying soil and land management conditions by applying three different adaptive techniques: general linear models (GLMs), artificial neural networks (ANNs), and regression trees (RTs). The crop yield data used in this research consisted of 720 maize yield indices from 11 different land management trials in southern Uganda. The GLM showed the poorest results in terms of modeling accuracy, prediction accuracy and model uncertainty, which might suggest its inability to model the non-linear causal relationships present in complex soil–land and crop-management interactions. The other two models show significantly higher prediction accuracy than the GLM.

Ilan Alon, Min Qi, Robert J. Sadowski (2001) [29], Henry C. Co, Rujirek Boosarawongse (2007) [30] compared artificial neural networks with winter exponential smoothing and the Box-Jenkins ARIMA model found that, across different forecasting periods and different forecasting horizons, the ANN performed the best. The ANN outperformed in the first period in which the economic conditions were relatively volatile and the ANN performed relatively well in predicting unseen data.

Chinna Srachoom (2007) [31] presented an application of artificial neural network for weather forecasting with the objective of properly designing and developing an artificial neural network for quick, accurate and short-term weather forecasting in the Muang district of Chiang Mai province. This program was developed by using the Neural Network Toolbox of the MATLAB application. It is the main program in building, training and testing neural networks for predicting the quantity of rain from the information on previous weather conditions and the resulting weight is used to enable users to conveniently use the information. The results of predicting the amount of rain quantity was found to have an accuracy of 77.29 percent.

Tassanawan Kaewsai and Supoj Nitsuwat (2009) [32] developed a movie recommender system by using the collaborative filtering technique and K-Mean (K-Mean is a fast, easy way that is the most popular for segmentation by cutting into sections and measuring the quality by square error value which would be suitable for segmentation data when each group has a similar size). Due to greater consumer demand, the manufacturers and distributors need to find products and services to satisfy customers. To facilitate the purchase of goods and services in an era of competition, a lot of information causing information overload problems that make accessing and receiving information to meet the requirements difficult or take longer to obtain the information required. Collaborative Filtering is a successful technique and has been popular in the development of recommendation systems. The above researchers studied the traditional recommender systems and found that if a user or product that wanted to recommend an increasing number, a long time will be required to make a prediction and influence user satisfaction. The researchers made a movie recommendation system using Collaborative Filtering and K-Mean, which is the data mining technique involving the division of users into subgroups before entering collaborative filtering to develop a movie recommendation system to effectively meet demands within a short period of time.

Yugal Kumar and G. Sahoo (2012) [33] focused on the fundamental concept of data mining, i.e. classification techniques. In this paper, Bayes Net, naïve Bayes, naïve Bayes Uptable, Multilayer perceptron, Voted perceptron and J48 classifiers were used for the classification of a data set. The performance of these classifiers was analyzed with the help of Mean Absolute Error, Root Mean-Squared Error and Time Taken to build the model and the results can be shown statistically as well as graphically from six different classifiers used for the classification of data. These techniques were applied on two datasets in which one of data sets contained one-tenth of instance and one-third attribute as compared to the other data set. The fundamental concept in taking two datasets was to analyze the performance of the discussed classifiers for both datasets. The results of the experiment were able to conclude that the performance of the J48 classifier/technique is better in comparison to the other classifiers/techniques.

Arihito Endo, Takeo Shibata, Hiroshi Tanaka (2007) [34] presented optimal models for predicting the survival rate of breast cancer patients over a period of five years. In 2002, this study examined 37,256 follow-up patients who had been diagnosed with breast cancer and registered in the SEER program from 1992 to 1997. The study implemented seven common algorithms (Logistic Regression model, Artificial Neural Network (ANN), Naive Bayes, Bayes Net, Decision Trees with naive Bayes, Decision Trees (ID3) and Decision Trees (J48)) besides the most widely used statistical method (Logistic Regression model) to develop the prediction models. The accuracy of the experiment was 85.8±0.2%, 84.5±1.4%, 83.9±0.2%, 83.9±0.2%, 84.2±0.2%, 82.3±0.2%, 85.6±0.2% for the Logistic Regression model, ANN, Naive Bayes, Bayes Net, Decision Trees with naive Bayes, ID3 and J48, respectively. Therefore, it can be concluded that in this study, the logistic regression model displayed the highest accuracy. The J48 had the highest sensitivity and the ANN had the highest specificity. The decision tree models tended to show high sensitivity and the Bayesian models were apt to show the accuracy escalating, finding that the optimal algorithm might be different by the predicted objects and dataset.

Krit Somkantha, Wilaiporn Kultangwattana and Worrawit Kultangwattana (2013) [35] suggested the development of personnel assessment system using Artificial Neural Network (ANN). The artificial intelligence technology was applied to increase the efficiency of personnel assessment system in the university. The experimental efficiency of personnel assessment method by using data from 2 assessor, which is expert of assessment and the actual data (Ground truth), to referenced for training personnel assessment system. The cross validation was used to test the efficiency of the techniques. The proposed method was compared with the Bayes method and the K-nearest neighbor method. From the experimental results, it was found that the proposed method had better efficiency than the compared methods. Our method can assess the result of personnel very efficiency. The experimental results from proposed method are very close to the assessor and it can assess the personnel faster than the assessor, which is believed that the proposed method can be a suitable approach for developing the efficiency of personnel assessment system for good performance. The proposed method can further be applied to any assessment problems.

According to previous research, ANN has been applied with MATLAB for forecasting results in many fields such as crop yield, rainfall forecasting and weather forecasting, etc. In addition, CF is another common method that is generally used in the Product Recommender System for books, movies and music in Amazon and CD-NOW. The decision tree is used for the classification of the breast cancer data set and predicting the survival rate of breast cancer patients. Each of the abovementioned research methods has different accuracies. Therefore, this research used these methods to analyze and compare the performance to determine which methods are most appropriate for application with this case study.

CHAPTER III
RESEARCH METHODOLOGY

3.1 Data

This research used book distribution data of Kledthai store, a book distribution center that performs the following duties: analysis of customer needs, ordering books and distributing books to customers which are activities carried out by retail stores. In this research, retail stores will be called "store" as in Figure 3.1, by using book distribution data during from 2009 to 2011 and consisting of 5,000 Books divided into 21 categories published by 216 publishers at a total of 10 retail stores. Tables 3.1 show the details of book the distribution system. The database for the system contains 5 tables on publishers, categories, products and sales, as shown in Table 3.1

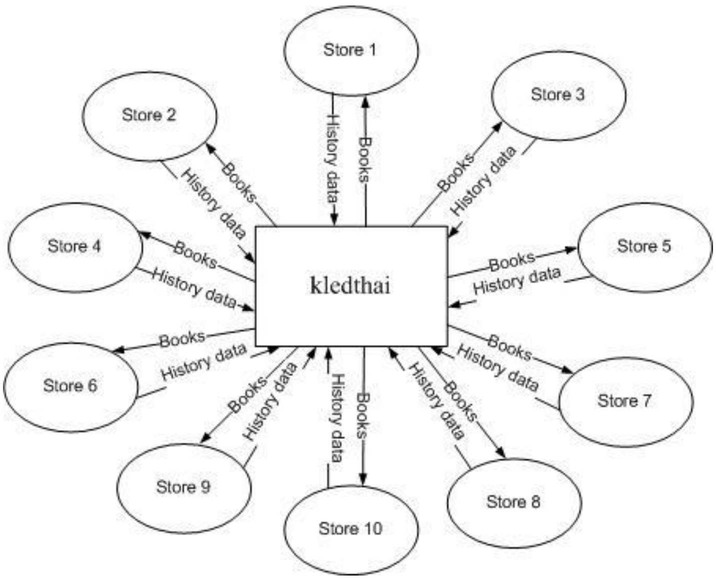


Figure 3.1 Kledthai Structure.

Table 3.1 The details of book distribution data.

store	publisher	categories
Store_id Store_name	Pub_id Pub_name	Cat_id Cat_name
Store data consisting of store codes and store names.	Publisher data consisting of publisher codes and publisher names.	Category data consisting of category codes and category names

product	sale
Product_id Product_name Price Released year Cat_id	Sale_id Store_id Product_id Pub_id Sale year Qty
Product data consisting of product codes, product names, product price, released year and category codes.	Distribution data consisting of sales codes, store codes, product codes, publisher codes, sale year and sales quantity

3.2 Pre-Analysis Data Preparation

1. Some sales quantity (Qty) are missing in which case, the value will be adjusted to zero before the analysis in order to imply that the store has no history of that product.
2. Publisher code appearing inconsistent will be converted to unique numerical codes.

3. The original sales quantity on the numbers sold each month which will be converted to order quantity data for each year and to total order quantity data for each book sold in the same year.
4. An additional year column was added to provide more information about the amount of period of the books released to the public and to show how many years the book is on the shelf.
5. With minimal data on the newly arrived books for quick distribution according to the adjustments identifying the best quantity for each bookstore. Precise predictions can vary greatly; therefore, the sales quantity is divided into 5 different classes as shown in 3.2, in order to know the approximate sales quantity of the given books. The 5 classes are “no” (unsalable), “low” (1-2 books can be sold), “medium” (3-4 books can be sold), “high” (5-7 books can be sold) and “veryhigh” (a high volume of books can be sold). This table suggests the sales trends for each store in selling newly arrived books.

Table 3.2 Shows the book prediction categories divided into 5 categories.

Classes	Circulation	Frequency
no	0	352
low	1 - 2	1256
medium	3 - 4	1128
high	5 - 7	1136
veryhigh	> = 8	1128

Table 3.3 Examples of book distribution data from before preparation.

Sale_id	Store_id	Product_id	Pub_id	Price	Sale month	Sale year	Released year	Qty
1	10301	9786115450114	2ก039	169	3	2010	2009	1
2	10725	9786115450190	B007	159	7	2009	2009	
3	10725	9786115450190	B007	159	5	2010	2009	4
4	10725	9786162260094	P005	169	9	2010	2009	10
5	20123	9786165450027	S-001	139	2	2010	2010	2
6	20123	9786165450027	S-001	139	6	2011	2010	
7	30121	9786167003436	ค020-1	290	8	2009	2009	6
8	50434	9786167059402	ค020-29	275	5	2011	2010	3
9	63420	9786115450114	ซ001	169	7	2009	2009	3
10	63420	9786115450114	ซ001	169	1	2010	2009	9
11	63420	9789741111329	ต017	150	9	2010	2009	15
12	83205	9789741640966	ท004	195	10	2011	2011	7
13	93301	9789748793658	ธ005	160	4	2010	2010	23
14	93302	9789749748510	พ039	135	2	2010	2010	3
15	93302	9789749748510	พ039	135	5	2011	2010	1

Sale_id is the sales code, Store_id is the store code, Product_id is the product code, Pub_id is the publisher code and Qty is the sales quantity.

Table 3.4 Example of book distribution data after data preparation.

Sale_id	Store_id	Product_id	Pub_id	Price	Sale month	Sale year	Released year	Year	Qty	Classes
1	10301	9786115450114	36	169	3	2010	2009	1	1	low
2	10725	9786115450190	8	159	7	2009	2009	0	0	no
3	10725	9786115450190	8	159	5	2011	2009	2	4	medium
4	10725	9786162260094	17	169	9	2010	2009	1	10	veryhigh
5	20123	9786165450027	22	139	2	2010	2010	0	2	low
6	20123	9786165450027	22	139	6	2011	2010	1	0	no
7	30121	9786167003436	87	290	8	2011	2009	2	6	high
8	50434	9786167059402	40	275	5	2011	2010	1	3	medium
9	63420	9786115450114	3	169	1	2009	2009	0	12	veryhigh
11	63420	9789741111329	26	150	9	2010	2009	1	15	veryhigh
12	83205	9789741640966	11	195	10	2011	2011	0	7	high
13	93301	9789748793658	3	160	4	2010	2010	0	23	veryhigh
14	93302	9789749748510	1	135	2	2010	2010	0	4	medium

3.3 Research Methodology

3.3.1 K-Fold Cross-Validation

The data will be partitioned using the K-fold cross-validation method. In k-fold cross-validation, the data is first partitioned into k equally (or nearly equally) sized segments or folds. Subsequently, the k iterations of training and validation are performed such that within each iteration a different fold of the data is held-out for validation while the remaining k - 1 folds are used for learning. The data is commonly stratified prior to being split into k folds. Stratification is the process of rearranging the data to ensure that each fold is a good representative of the whole [24]. In this study, 10-fold cross-validation will be used for the examples in Figure 3.2 showing how the input data was partitioned by using 10-fold cross validation.

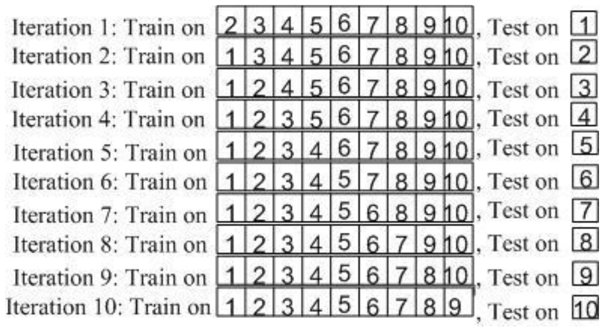


Figure 3.2 Shows how the input data was partitioned by using 10-fold cross validation.

3.3.2 Input Data

Data from ten different book stores have been used. Each store had a total of 500 samples of book distribution data. This data are divided into training data and testing data by the 10-fold cross validation method. The attributes of each input are category code (Cat_id), publisher code (Pub_id), price and year. The sales quantity is the prediction value which is divided into 5 classes as shown in Table 3.4 which shows book prediction categories to be divided into 5 classes**.**

The inputs used in the experiment were divided into 2 formats: numeric data consists of numbers only, can be calculated, and nominal data, indicating a separate category in the data that have no numerical meaning, cannot be calculated. Thus, each technique was selected. The experiment will be performed by using these 2 formats. In case of Nominal data, Data type of Cat_id and Pub_id are Nominal data. Another case, Data type of Cat_id and Pub_id are numeric data.

3.3.3 Examining the importance of Attributes

In this research, each book consist of 4 attributes, Categories data (Cat_id), Publisher data (Pub_id), Price and Year. Therefore, in this part will uses these attributes as input data, to analyze the significance of each attribute by considering accuracy value. For comparison, 3 input pattern are used: input data is Cat_id only, input data is Pub_id only, and input data are Pub_id and Cat_id, will analyzed separately for each store, which step of each case as follows:

- Input data is Cat_id.

  (1) Sort Cat_id by sorting from maximum sale quantity value, (2) Calculate the average of sale quantity value of each Cat_id, this value

will be used for representing prediction value of sales quantity of the book that classified in those Cat_id, (3) Converted those averages to classes by reference from Table 3.2, for example in Table 3.5, to create confusion matrix and calculate accuracy value. For this case, training data and testing data used are the same data set.

Table 3.5 Example of input data that use Cat_id.

Cat_id	Qty	Mean of Qty	Class
11	525	7.291666667	high
2	476	6.102564103	high
1	467	5.626506024	high
6	322	21.46666667	veryhigh
8	319	10.29032258	veryhigh
7	294	8.166666667	veryhigh
9	274	6.85	high
4	212	5.888888889	high
02	140	35	veryhigh
10	125	31.25	veryhigh
21	110	5.5	high
31	97	13.85714286	veryhigh

- Input data is Pub_id.

  (1) Sort Pub _id by sorting from maximum sale quantity value, (2) Calculate the average of sale quantity value of each Pub _id, this value will be used for representing prediction value of sales quantity of the book that classified in those Pub _id, (3) Converted those averages to classes by reference from Table 3.2, for example in Table 3.6, to create confusion matrix and calculate accuracy value. For this case, training data and testing data used are the same data set.

Table 3.6 Example of input data that use Pub_id.

Pub_id	Qty	Mean of Qty	Class
66	576	23.04	veryhigh
21	285	35.625	veryhigh
69	180	4.186046512	medium
87	169	8.047619048	veryhigh
223	140	35	veryhigh
1	125	5.681818182	high
24	113	16.14285714	veryhigh
100	109	18.16666667	veryhigh
68	106	7.066666667	high
102	98	24.5	veryhigh
92	86	6.615384615	high
49	83	5.928571429	high

- Input data is Pub_id and Cat_id.

  (1) Sort Pub_id and Cat_id by sorting from maximum sale quantity value, (2) Calculate the average of sale quantity value of each Pub_id and Cat_id, this value will be used for representing prediction value of sales quantity of the book that classified in those Pub_id and Cat_id, (3) Converted those averages to classes by reference from Table 3.2, for example in Table 3.7, to create confusion matrix and calculate accuracy value. For this case, training data and testing data used are the same data set.

Table 3.7 Example of input data that use Pub_id and Cat_id.

Pub_id	Cat_id	Qty	Mean of Qty	Class
66	6	182	22.75	veryhigh
66	2	118	19.66666667	veryhigh
66	11	97	19.4	veryhigh
66	31	71	35.5	veryhigh
66	5	60	30	veryhigh
66	9	48	24	veryhigh
21	10	108	54	veryhigh
21	7	93	23.25	veryhigh
21	11	84	42	veryhigh
223	02	140	35	veryhigh
54	4	33	6.6	high
54	11	24	8	veryhigh
136	9	22	3.666666667	medium
136	1	20	3.333333333	medium

3.3.4 Collaborative Filtering (CF)

Concerning the category codes (Cat_id), publisher codes (Pub_id), price and years, the price and years are the main data used for each book item. All attributes are normalized before use by equations (15) and (17) as follows:

$$Price_{dif} = Price_{max} - Price_{min} \quad (3.1)$$

$$Price_{n} = \frac{Price}{Price_{dif}} \quad (3.2)$$

$$Year_{dif} = Year_{max} - Year_{min} \quad (3.3)$$

$$Year_{n} = \frac{Year}{Year_{dif}} \quad (3.4)$$

where $Price_{max}$ is the most expensive price, $Price_{min}$ is the cheapest price, $Price$ is the original price, $Price_{n}$ is the price that was normalized, $Year_{max}$ is the highest year, $Year_{min}$ is the lowest year, $Year$ is the original year, $Year_{n}$ is the year that was normalized.

Table 3.8 Example of the normalized input data used in the correlative filtering method.

Product_id	Cat_id	Pub_id	Price	Year
9786119002883	25	34	0.079476861	0.1
9786119002883	25	34	0.079476861	0
9786167003078	31	69	0.075452716	0
9786167003078	31	69	0.075452716	0.1
9786167003085	31	69	0.080482897	0
9786167003085	31	69	0.080482897	0.1
9786167003139	9	69	0.100603622	0.1
9786167003139	9	69	0.100603622	0
9786167102009	1	68	0.085513078	0.1
9786167102009	1	68	0.085513078	0
9786167147017	5	66	0.082997988	0.1
9786167147017	5	66	0.082997988	0
9786167147116	6	66	0.098088531	0
9786167147116	6	66	0.098088531	0.1

Table 3.8 shows the input data used in the collaborative filtering method in which price and year data was normalized, including the step of calculating prediction values as follows:

Calculating the Predicted Value

1. Calculate the similarities among the attributes of the book by calculating from the category codes (Cat_id), publisher codes (Pub_id), price and year in which the standard Pearson r Correlation Coefficient was used to measure the similarity attributes of the book.
2. Select nine books with the highest similarities with the active book, which is commonly called the neighboring and will be called neighboring book in this research. High degrees of similarity were selected in only 9 books because, although more could have been selected, the results would be no different, selecting more books would have required more time for analysis.
3. Calculate the weights between the active books and neighboring books, by using the distance from the active book.
4. Compute a prediction from a weighted combination of the selected neighbors' ratings, using (2).

The results with the previous method, the actual value and predicted value were obtained from the calculations are shown in Table 3.9. The predicted value is the predicted value of the sales quantity (Qty) with reference to Table 3.2 showing the book prediction categories converted into 5 classes. This case, the data is divided into training data and testing data by using 10 fold cross validation method

Table 3.9 Example of the results of prediction by the correlative filtering method.

Actual		Predict	
Qty	Class	Qty	Class
22	veryhigh	9.91	veryhigh
2	low	4.56	medium
9	veryhigh	10.67	veryhigh
3	medium	4.11	medium
0	no	2.33	low

3.3.5 Artificial Neural Network (ANN)

Artificial neural networks are a technique in artificial intelligence. Multi-Layer Perceptron forms one type of a neural network. A multilayer perceptron is a feed-forward artificial neural network model consisting of a system of simple interconnected neurons, or nodes representing a nonlinear mapping between an input vector and an output vector. The nodes are connected by weights and output signals which are a function of the sum of the inputs to the node modified by a simple nonlinear transfer or activation function as shown in Figure 3.3. Multilayer perceptron utilize a supervised learning technique called back-propagation for training the network.

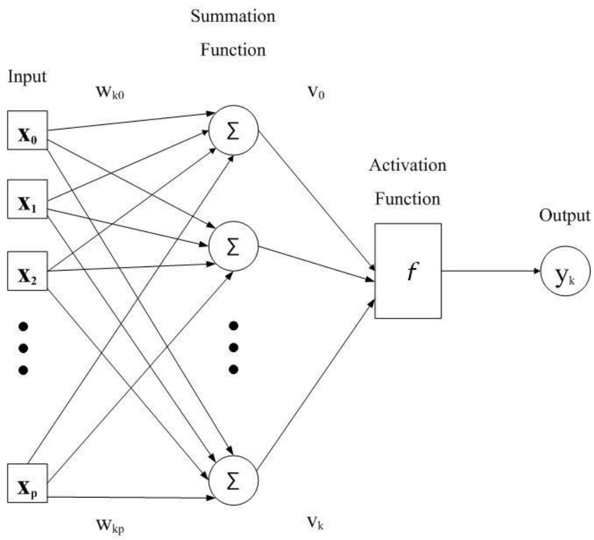


Figure 3.3 Structure of Multilayered Perceptron.

In this research, use hyperbolic tangent function as activation function, this function is a good tradeoff for neural networks, where speed is important and the exact shape of the transfer function is not. The hidden unit will be set to 11 for input data is numeric data, because the experiment uses a different hidden unit. Using the hidden unit at 11 shows the best predictions according to Table 3.10, which shows the average accuracy of each hidden unit from 5 – 13 hidden units, while the hidden unit at 40 for nominal input data is used, because the experiment uses a different input values. Using the hidden unit at 40 shows the best predictions according to Table 3.11, which shows the accuracy of each hidden unit from 36 to 50 hidden units.

Table 3.10 The accuracy of each hidden unit for input data is numeric data.

10 fold	Hidden Units				
	5	7	9	11	13
1	0.22	0.22	0.22	0.22	0.2
2	0.24	0.32	0.26	0.34	0.32
3	0.34	0.24	0.36	0.34	0.32
4	0.26	0.3	0.28	0.22	0.26
5	0.24	0.2	0.38	0.3	0.26
6	0.24	0.28	0.24	0.3	0.32
7	0.22	0.18	0.24	0.2	0.18
8	0.26	0.28	0.32	0.4	0.34
9	0.26	0.22	0.36	0.24	0.26
10	0.28	0.26	0.28	0.38	0.26
Mean	***0.256***	***0.25***	***0.294***	***0.294***	***0.272***
S.D.	***0.04***	***0.05***	***0.06***	***0.07***	***0.05***

Table 3.10 shows the accuracy from the experiment using the numbers of hidden units at 5, 7, 9, 11, and 13 by using 10 fold cross validation. According to the experiment, the numbers of hidden units yielding the best accuracy value are 9 and 11. Hence, both hidden unit values were tested with the data from the other remaining nine stores. According to the results, the number of 11 hidden units yielded the best accuracy suitable for this case study. In this research, therefore, the hidden unit = 11 was used.

Table 3.11 The accuracy of each hidden unit for input data is nominal data.

10 fold	Hidden Units							
	36	38	40	42	44	46	48	50
1	0.31	0.33	0.35	0.34	0.35	0.34	0.33	0.32
2	0.31	0.31	0.34	0.33	0.31	0.31	0.31	0.31
3	0.28	0.28	0.22	0.27	0.24	0.24	0.25	0.26
4	0.30	0.28	0.31	0.29	0.27	0.32	0.29	0.31
5	0.29	0.32	0.31	0.33	0.32	0.31	0.28	0.31
6	0.24	0.24	0.25	0.22	0.22	0.26	0.22	0.25
7	0.42	0.40	0.45	0.42	0.46	0.46	0.45	0.43
8	0.34	0.36	0.36	0.36	0.34	0.33	0.36	0.36
9	0.47	0.46	0.49	0.43	0.47	0.47	0.49	0.46
10	0.58	0.59	0.59	0.59	0.62	0.59	0.61	0.59
Mean	***0.355***	***0.356***	***0.367***	***0.358***	***0.360***	***0.363***	***0.359***	***0.360***
S.D.	***0.10***	***0.10***	***0.11***	***0.10***	***0.12***	***0.11***	***0.12***	***0.11***

Table 3.11 shows the accuracy from the experiment using the numbers of hidden units at 36, 38, 40, 42, 44, 46, 48 and 50 by running 10 fold cross validation.

According to the experiment, the numbers of hidden units yielding the best accuracy value are 40. In this research, therefore, the hidden unit = 40 was used. The accuracy on the test data was used to find the most appropriate complexity for the best generalization and avoid over fitting of the model.

- Numeric Data

Input node consist of 4 nodes are Cat_id, Pub_id, Price and Year, is numeric data. For example in Table 3.12, Output node consist of 1 node is prediction value of sales quantity (Qty).

Table 3.12 Example of the input data used in the artificial neural network method.

Cat_id	Pub_id	Price	Year
25	34	158	1
25	34	158	0
31	69	150	0
31	69	150	1
31	69	160	0
31	69	160	1
9	69	200	1
9	69	200	0
1	68	170	1
1	68	170	0
5	66	165	1
5	66	165	0
6	66	195	0

The results with the previous method, the actual value and predicted value will be converted into a form of class by reference from Table 3.2, for example in Table 3.13, then display result in form of confusion matrix, to determine the accuracy further.

Table 3.13 Example of the results of prediction by the artificial neural network method.

Book id	Cat_id	Pub_id	Price	Year	Actual Qty	Predict Qty	Actual Class	Predict Class
75	14	53	150	0	0	3.5185	no	medium
162	1	47	100	1	10	6.8619	veryhigh	high
387	9	111	100	9	3	6.9059	medium	high
391	23	125	65	6	6	7.3797	high	high

- Nominal Data

Input node consist of 243 nodes are Cat_id with 25 nodes, Pub_id with 216 nodes, Price with 1 node and Year with 1 node which data type of Cat_id and Pub_id are nominal data. The output node consisting of 5 nodes is the prediction value of sales quantity (Qty) in a form of different probabilities of 5 different classes, according to Table 3.14

Table 3.14 Shows 5 output node of neural network.

Classes	Circulation	5 output node
no	0	10000
low	1 - 2	01000
medium	3 - 4	00100
high	5 - 7	00010
veryhigh	> = 8	00001

3.3.6 Decision Tree

Decision trees are used to build predictive models from existing data in the form of a tree. The decision tree is a supervised learning method, which can create a classification model automatically from a sample of data set in advance and called a training set capable of making to predictions for a group of items that have not been categorized.

The decision tree determines the most important attribute and split its leaves by the selected attributes. The split leaves, therefore, will keep selecting the most important of the branches in a hieratical order.

The J48 algorithm was used in our work where the confidence factor used for pruning (smaller values incur more pruning) has set at 0.25.

- Numeric Data

  The input data consisted of the category code (Cat_id), publisher code (Pub_id), Price, Year, which data type of input data is numeric data. Target value is sales quantity (Qty), was converted by the data in Table 3.2., for example in Table 3.15 shows examples of input data and target data of the decision tree by the J48 algorithm.

Table 3.15 Examples of Input Data of J48 Algorithm.

Cat_id	Pub_id	Price	Year	Qty
25	34	158	1	no
25	34	158	0	medium
31	69	150	0	no
31	69	150	1	high
31	69	160	0	no
31	69	160	1	high
9	69	200	1	no
9	69	200	0	high
1	68	170	1	no
1	68	170	0	high
5	66	165	1	no
5	66	165	0	veryhigh
6	66	195	0	no

The J48 algorithm will be used to help decide in creating a decision tree as shown in Figure 3.4 - Example of a Decision Tree

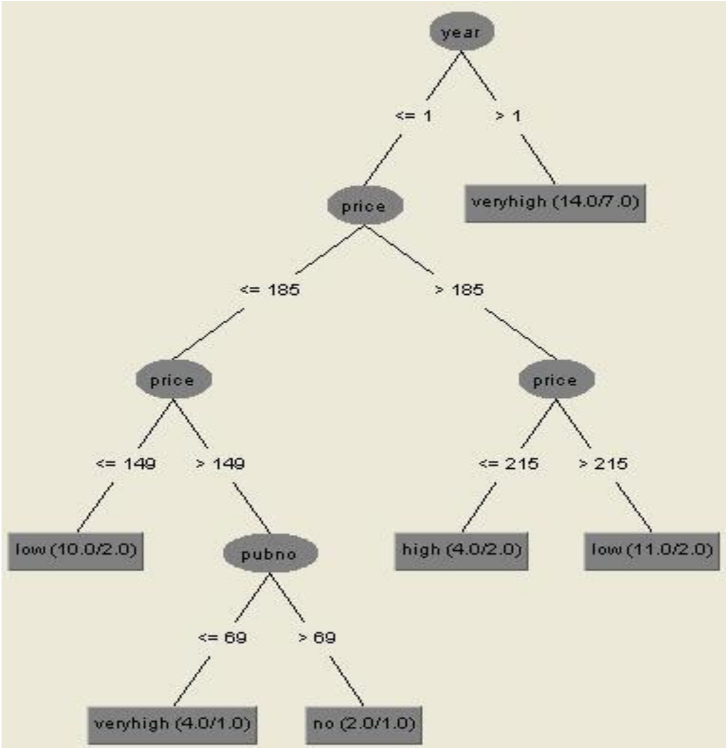


Figure 3.4 Example of Decision Tree of Store ID 10301.

Figure 3.4 shows an example decision tree of Store ID 10301, which can be described as follows: If *year > 1*, sales trends are high or represent a "veryhigh" class which predicts a "veryhigh" class at a total of 14 volumes in which 7 of 14 books, or 50%, are correctly predicted, the suggestion is that half of the books in the bookstore for more than a year can be sold in a very high volume, by not using other attributes.

At the same time the sales trends for books at *year <= 1* will depend on the price. If the *price is >215* representing that the book is in a “low” class with only 2 out of 11, or 18%, the prediction is incorrect. On the other hand, if the *price is <= 149* representing that the book is in a “low” class, the suggestion is that most of the books can be sold at a "low" level, the bracket shows that only 2 out of 10 were predicted incorrectly, thereby suggesting that approximately 80% of the books with a price range of more than 215 baht or priced less than or equal to 149 baht can be predicted correctly by using the year and price alone.

However, if the *price is > 185* baht but the *price is <=215* baht represents the book to be in a “high” class which predicts is the book to be in a "high" class a total of 4 books, 2 of which were predicted incorrectly.

However, if the *price is > 149* baht but *price <=185* baht, the prediction will depend on the pubno. If *pubno > 69* that is in a “no” class which predicts the book to be in a "no" class for a total of 2 books, 1 of which was predicted incorrectly. On the other hand, if the *pubno is <= 69*, the book is represented as being in a “veryhigh” class which predicts is the book to be in a “veryhigh" class for a total of 4 books, 1 of which was predicted incorrectly.

The outcome indicates that with some particular set of attributes we can more certainly predict the tendency of the sale than some other test of attributes. This method therefore can easily give hierarchical importance to different combinations of attributes that is unique from the other methods.

This input data can be done if only Cat_id and Pub_id are organized neatly that the closer the value of Cat_id or Pub_id refer to similar categories or publisher. However, this method can be dangerously interpreted. The research has raised this method only for comparison and the Cat_id or Pub_id are partly organized for this numeric data. The interpretation, however, has to be carefully considered.

- Nominal Data

  The input data consisted of the category code (Cat_id), publisher code (Pub_id), Price, Year, which data type of Cat_id and Pub_id are nominal data. The results are displayed in form of decision tree as well. But, because of tree resulting from this data type is too large and too complex, to understand, making it difficult for analysis, and make decisions. The experiments using the 3 discussed methods will be done in the Chapter 4.

CHAPTER IV
RESULTS

4.1 Performance Measures

The experiments in this research were evaluated by using the standard accuracy, precision (specification) and recall (sensitivity) methods. These were calculated by using the predictive classification table known as Table 4.1 Confusion Matrix (35, 36).

Table 4.1 Confusion Matrix.

		PREDICTED	
		IRRELEVANT	RELEVANT
ACTUAL	IRRELEVANT	**TN**	**FP**
	RELEVANT	**FN**	**TP**

Considering Table 4.1:

TN (True Negative) is the number of correct predictions in which an instance is *irrelevant*

FP (False Positive) is the number of incorrect predictions in which an instance is *relevant*

FN (False Negative) is the number of incorrect predictions in which an instance is *irrelevant*

TP (True Positive) is the number of correct predictions in which an instance is *relevant*

Accuracy – The proportion of the total number of predictions that are correct:

Accuracy = (TN + TP) / (TN + FN + FP + TP) (4.1)

Precision – The proportion of the predicted *relevant* pages that are correct:

Precision = TP / (FP + TP) (4.2)

Recall – The proportion of the *relevant* pages that are correctly identified

Recall = TP / (FN + TP) (4.3)

4.2 Examining the importance of Attributes

From experiment to determine the significance of each attribute which will estimate of accuracy value. Divided into 3 cases are Input data is Cat_id only, Input data is Pub_id only, and Input data are Pub_id and Cat_id. Show the results of each case as following:

4.2.1 Input data is Cat_id

Table 4.2 Confusion matrix of using Cat_id as input data.

		PREDICTED						
	CF	no	low	medium	high	veryhigh	Recall (%)	sum
ACTUAL	no	10	161	58	68	55	3	352
	low	2	272	285	453	244	22	1256
	medium	0	161	296	464	207	26	1128
	high	0	92	248	500	296	44	1136
	veryhigh	0	10	118	483	517	46	1128
	Precision (%)	83	39	29	25	39	28.1 / 43.3	
	sum	12	696	1005	1968	1319		Average (%)
	Accuracy (%)							31.90%

Table 4.2 shows the confusion matrix of 5,000 books from 10 different stores by using Cat_id as input data. From the experiment by using Cat_id as input data, can accurately predict at 31.90% which a "no" class has the highest precision and the lowest recall at 83% and 3%, respectively. because of the number of the actual books and the books predicted in the "no" class has 12 and 352 books, respectively which this case can accurately predicted in this class total 10 books. On the other hand, the class that has the highest recall and the lowest precision is a "veryhigh" class at 46%, the number of the actual books in a "veryhigh" class has 1128 books which can accurately predicted at a total 517 books, and a "high" class at 25%, the number of the books predicted in the “high” class has 1968 books which can accurately predicted at a total 500 books, respectively.

4.2.2 Input data is Pub_id

Table 4.3 Confusion matrix of using Pub_id as input data.

		PREDICTED						
	CF	no	low	medium	high	veryhigh	Recall (%)	sum
ACTUAL	no	59	189	38	16	50	17	352
	low	1	416	334	269	236	33	1256
	medium	0	199	516	250	163	46	1128
	high	0	83	378	436	239	38	1136
	veryhigh	0	9	104	377	638	57	1128
	Precision (%)	98	46	38	32	48	38.1 / 52.6	
	sum	60	896	1370	1348	1326		Average (%)
	Accuracy (%)							41.30%

Table 4.3 shows the confusion matrix of 5,000 books from 10 different stores by using Pub_id as input data. From the experiment by using Pub_id as input data, can accurately predict at 41.30% which a "no" class has the highest precision and the lowest recall at 98% and 17%, respectively. because of the number of the actual books and the books predicted in the "no" class has 60 and 352 books, respectively which this case can accurately predicted in this class total 59 books. On the other hand, the class that has the highest recall and the lowest precision is a "veryhigh" class at 57%, the number of the actual books in a "veryhigh" class has 1128 books which can accurately predicted at a total 638 books, and a "high" class at 32%, the number of the books predicted in the “high” class has 1348 books which can accurately predicted at a total 436 books, respectively.

4.2.3 Input data are Pub_id and Cat_id

Table 4.4 Confusion matrix of using Pub_id and Cat_id as input data.

		PREDICTED						
	CF	no	low	medium	high	veryhigh	Recall (%)	sum
ACTUAL	no	84	176	32	18	42	24	352
	low	1	454	345	250	206	36	1256
	medium	0	209	533	250	136	47	1128
	high	0	69	380	484	203	43	1136
	veryhigh	0	7	79	363	679	60	1128
	Precision (%)	99	50	39	35	54	42 55.3	
	sum	85	915	1369	1365	1266		Average (%)
	Accuracy (%)							44.68%

Table 4.4 shows the confusion matrix of 5,000 books from 10 different stores by using Pub_id and Cat_id as input data. From the experiment by using Pub_id and Cat_id as input data, can accurately predict at 44.68% which a "no" class has the highest precision and the lowest recall at 99% and 24%, respectively. Because of the number of the actual books and the books predicted in the "no" class has 85 and 352 books, respectively which this case can accurately predicted in this class total 84 books. On the other hand, the class that has the highest recall and the lowest precision is a "veryhigh" class at 60%, the number of the actual books in a "veryhigh" class has 1128 books which can accurately predicted at a total 679 books, and a "high" class at 35%, the number of the books predicted in the “high” class has 1365 books which can accurately predicted at a total 484 books, respectively.

From the results of the experimenting to find importance of attributes, concluded that Pub_id are more important than Cat_id because of the experiment using Pub_id as input data are more accuracy than using Cat_id as input data. But when experimenting by used Pub_id and Cat_id as input data, are the most accuracy because some books not able to predict by using only Cat_id or Pub_id.

4.3 Numeric data

Numeric data consists of real numbers only and can be calculated.

4.3.1 Artificial Neural Network (ANN)

In this research, the Multi-Layer Perceptron algorithm and hyperbolic tangent function are used as activation function. The numbers of hidden units were set to 11. The input nodes consist of 4 nodes: Cat_id, Pub_id, Price and Year, all of which is numeric data. The expected outcome would be represented in the form of a confusion matrix to find precision, recall and accuracy as shown in Table 4.5

Table 4.5 Confusion matrix of the ANN method which data type of input data is numeric data.

		PREDICTED						
	ANN	no	low	medium	high	veryhigh	Recall (%)	sum
ACTUAL	no	25	163	74	47	43	7	352
	low	71	314	280	299	292	25	1256
	medium	52	254	257	307	258	23	1128
	high	54	156	214	341	371	30	1136
	veryhigh	27	69	128	322	582	52	1128
	Precision (%)	11	33	27	26	38	27 / 27.4	
	sum	229	956	953	1316	1546		Average (%)
	Accuracy (%)							30.38%

Table 4.5 shows the confusion matrix of 5,000 books from 10 different stores by the ANN method. From this table, books predicted in the “no” class at a total of 229 books, while the ANN method can accurately predict only 25 books which this class has the lowest precision at 11% when compared to the other class. When considered in term of recall, the “no” class has the lowest recall as well, at 7%. On the other hand, the “veryhigh” class has the highest recall at 52%, ANN method can accurately predict 582 books of the actual value in this class at a total 1128 books. In addition, the "veryhigh" class has the highest precision at 38% because of the books predicted in the “veryhigh” class at a total of 1546 books, while the ANN method can accurately predict 582 books.

The experimental results yielded by the ANN method have an overall accuracy of 30.38%.

4.3.2 Decision Tree

The decision tree method used a J48 algorithm where the confidence factor was set at 0.25. The input data consisted of the Cat_id, Pub_id, Price, and Year, which data type of input data is numeric data. Target value is sales quantity (Qty), was converted to class form.

Decision trees involving the creation of a tree to help decide based on the given inputs. The decision tree method ignores the weight of the input data. In other words, the amount of data in each class is different and rarely effects, but will consider conditions, respectively. The advantage of the decision tree is that the attributes are divided based on the importance of explicitly. In this method, therefore, the outcome prediction will not only be given, it will serve as a guide on the importance of each attribute in a hierarchical order. The expected outcome in terms of accuracy would be represented in a form confusion matrix to determine the precision, recall and accuracy values as shown in Table 4.6

Table 4.6 Confusion matrix of the decision tree method which data type of input data is numeric data.

		PREDICTED						
	Decision tree	no	low	medium	high	veryhigh	Recall (%)	sum
ACTUAL	no	170	68	42	38	34	48	352
ACTUAL	low	64	558	264	209	161	44	1256
ACTUAL	medium	67	313	368	233	147	33	1128
ACTUAL	high	59	226	262	338	251	30	1136
ACTUAL	veryhigh	26	176	136	219	571	51	1128
	Precision (%)	44	42	34	33	49	40.4 / 41.2	
	sum	386	1341	1072	1037	1164		Average (%)
	Accuracy (%)							40.10%

Table 4.6 shows the confusion matrix of 5,000 books from 10 different stores by the decision tree method where the “high” class had the lowest recall at 30%.

In this case study, the actual value is in the "high" class with a total of 1,136 books. At the same time, the decision tree method can accurately predict the “no” class for only 338 books of the aforementioned. On the other hand, the “veryhigh” class had the highest recall at 51%. The actual value is in a “veryhigh” class at a total of 1,128 books, while the decision tree method can accurately predict 571 books of the aforementioned. When considered in terms of precision, the decision tree method had a precision of the “high” class at 33%. The books predicted in a “high” class at a total of 1,037 books, while the decision tree method can accurately predict 338 books of the aforementioned. On the other hand, the “veryhigh” class had the lowest precision at 49%. The books predicted in a “veryhigh” class at a total of 1,164 books, while the decision tree method can accurately predict 571 books of the aforementioned.

The experimental results yielded by the decision tree method had an overall accuracy of 40.10%. The example of the experiment results yielded by using the decision tree method in Figures 4.1-4.3 is shown in the form of part of a tree. The decision tree method is created from data analyzed for the 3 stores raised as examples. The actual tree structure yielded by the analysis has more detailed branches, but the researcher has summarized to ensure that only the first three remain for the tree structure which can view the full decision tree at Appendices. The lowest branches give the overall accuracy of each branch. For example, If *Year <= 0*, *Price <= 80* and *Pub_id <= 76*, overall accuracy was 69%.

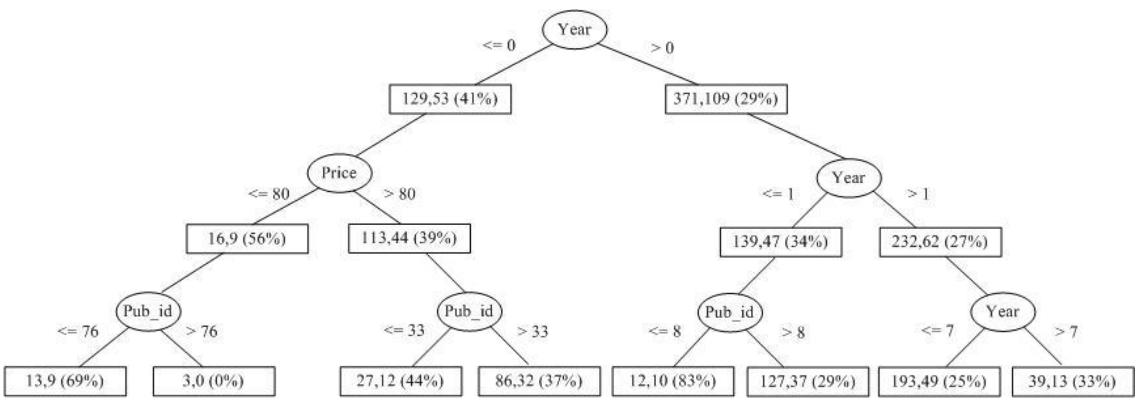


Figure 4.1 Example of the Decision Tree for Store ID 11218.

Table 4.7 Detail of the Decision Tree for Store ID 11218.

Condition	Total Predict,Predict Correct	no	low	medium	high	veryhigh
Year <= 0	129,53(41%)	14.7%	9.3%	15.5%	2.0%	0.0%
Price <= 80	16,9(56%)	0.0%	50.0%	6.3%	0.0%	0.0%
Pub_id <= 76	13,9(69%)	0.0%	61.5%	7.7%	0.0%	0.0%
Pub_id > 76	3,0(0%)	0.0%	0.0%	0.0%	0.0%	0.0%
Price > 80	113,44(39%)	16.8%	3.5%	16.8%	1.8%	0.0%
Pub_id <= 33	27,12(44%)	0.0%	11.1%	33.3%	0.0%	0.0%
Pub_id > 33	86,32(37%)	22.1%	1.2%	11.6%	2.3%	0.0%
Year > 0	371,109(29%)	0.8%	4.9%	1.9%	8.6%	13.2%
Year <= 1	139,47(34%)	1.4%	5.0%	2.2%	15.1%	10.1%
Pub_id <= 8	12,10(83%)	0.0%	0.0%	0.0%	0.0%	83.3%
Pub_id > 8	127,37(29%)	1.6%	5.5%	2.4%	16.5%	3.1%
Year > 1	232,62(27%)	0.4%	4.7%	1.7%	4.7%	15.1%
Year <= 7	193,49(25%)	0.5%	5.2%	2.1%	5.2%	12.4%
Year > 7	39,13(33%)	0.0%	2.6%	0.0%	2.6%	28.2%

Table 4.7 shows detail of the Decision Tree of Store ID 11218. For example: If *Year <= 0*, the opportunity to predicted correctly was 41%, 14.7% of a "no" class, 9.3% of a "low" class, 15.5% of a "medium" class and 2.0% of a "high" class which don't have outstanding class. But If *Year <= 0* and *Price <= 80*, will increase the opportunity to predicted correctly was 56%, most of come from a "low" class to 50%, represent that if book was put up for sold less than 1 year and price is less than or equal to 80, have the opportunity to sold less or a "low" class.

Table 4.8 Confusion Matrix of 500 books from Store ID 11218.

11218	no	low	medium	high	veryhigh	sum
no	23	12	7	8	7	57
low	4	33	29	30	16	112
medium	10	29	32	26	13	110
high	9	27	28	21	25	110
veryhigh	5	24	16	23	43	111
sum	51	125	112	108	104	

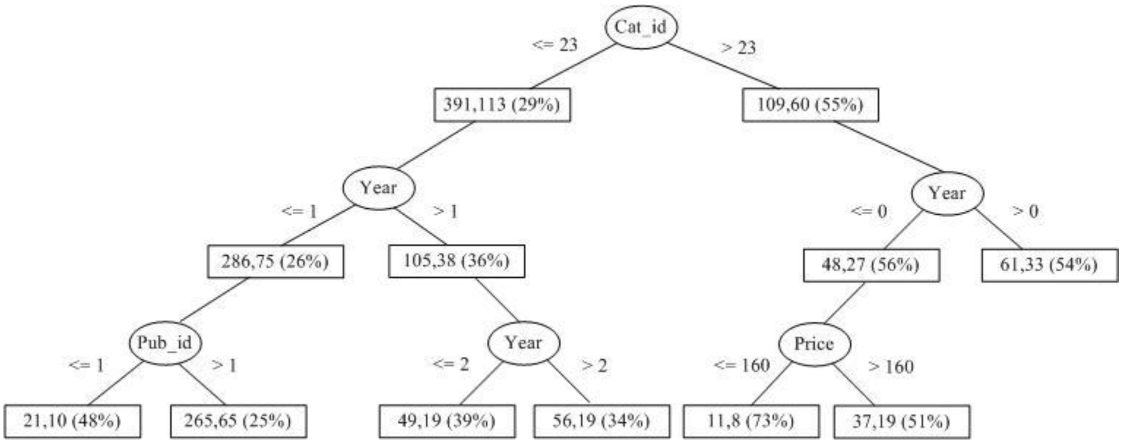


Figure 4.2 Example of the Decision Tree for Store ID 30121.

Table 4.9 Detail of the Decision Tree for Store ID 30121.

Condition	Total Predict,Predict Correct	no	low	medium	high	veryhigh
Cat_id <= 23	391,113(29%)	3.6%	3.3%	7.4%	5.9%	8.7%
Year <= 1	286,75(26%)	4.9%	1.0%	4.2%	6.6%	9.4%
Pub_id <= 1	21,10(48%)	42.9%	4.8%	0.0%	0.0%	0.0%
Pub_id > 1	265,65(25%)	1.9%	0.8%	4.5%	7.2%	10.2%
Year > 1	105,38(36%)	0.0%	9.5%	16.2%	3.8%	6.7%
Year <= 2	49,19(39%)	0.0%	16.3%	16.3%	6.1%	0.0%
Year > 2	56,19(34%)	0.0%	3.6%	16.1%	1.8%	12.5%
Cat_id > 23	109,60(55%)	47.7%	7.3%	0.0%	0.0%	0.0%
Year <= 0	48,27(56%)	39.0%	16.7%	0.0%	0.0%	0.0%
Price <= 160	11,8(73%)	0.0%	72.7%	0.0%	0.0%	0.0%
Price > 160	37,19(51%)	51.4%	0.0%	0.0%	0.0%	0.0%
Year > 0	61,33(54%)	54.1%	0.0%	0.0%	0.0%	0.0%

Table 4.9 shows detail of the Decision Tree of Store ID 30121. For example: If *Cat_id > 23*, the opportunity to predicted correctly was 55%, most of come from a “no” class to 47.7%, represent that if categories code of book have more than 23, have the opportunity to unsaleable.

Table 4.10 Confusion Matrix of 500 books from Store ID 30121.

30121	no	low	medium	high	veryhigh	sum
no	66	7	10	12	5	100
low	21	21	19	22	17	100
medium	20	14	29	24	13	100
high	19	19	20	23	19	100
veryhigh	10	18	18	20	34	100
sum	136	79	96	101	88	

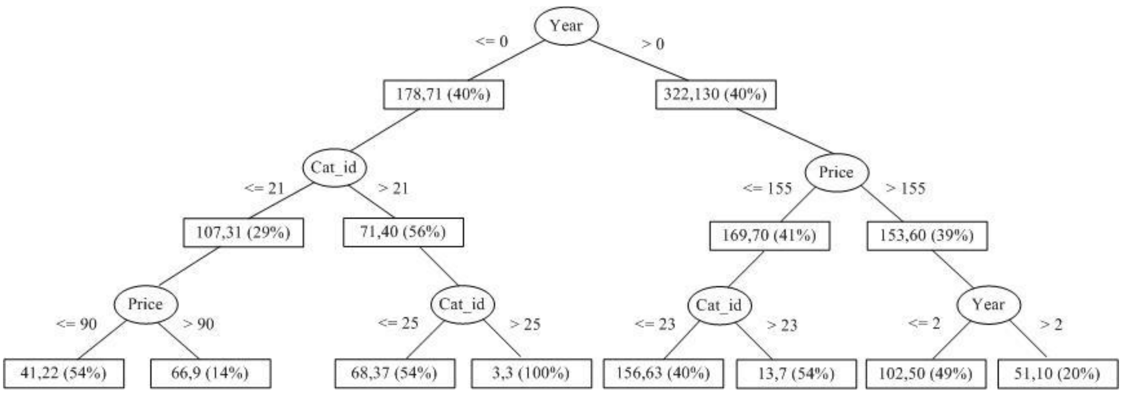


Figure 4.3 Example of the Decision Tree for Store ID 83205.

Table 4.11 Detail of the Decision Tree for Store ID 83205.

Condition	Total Predict,Predict Correct	no	low	medium	high	veryhigh
Year <= 0	178,71(40%)	0.0%	2.8%	21.3%	6.7%	9.0%
Cat_id <= 21	107,31(29%)	0.0%	1.9%	0.9%	11.2%	15.0%
Price <= 90	41,22(54%)	0.0%	2.4%	0.0%	14.6%	36.6%
Price > 90	66,9(14%)	0.0%	1.5%	1.5%	9.1%	1.5%
Cat_id > 21	71,40(56%)	0.0%	4.2%	52.1%	0.0%	0.0%
Cat_id <= 25	68,37(54%)	0.0%	0.0%	54.4%	0.0%	0.0%
Cat_id > 25	3,3(100%)	0.0%	100.0%	0.0%	0.0%	0.0%
Year > 0	322,130(40%)	0.0%	28.9%	1.6%	7.5%	2.5%
Price <= 155	169,70(41%)	0.0%	20.7%	2.4%	14.2%	4.1%
Cat_id <= 23	156,63(40%)	0.0%	19.2%	2.6%	15.4%	3.2%
Cat_id > 23	13,7(54%)	0.0%	38.5%	0.0%	0.0%	15.4%
Price > 155	153,60(39%)	0.0%	37.9%	0.7%	0.0%	0.7%
Year <= 2	102,50(49%)	0.0%	48.0%	0.0%	0.0%	1.0%
Year > 2	51,10(20%)	0.0%	17.6%	2.0%	0.0%	0.0%

Table 4.11 shows detail of the Decision Tree of Store ID 83205. For example: If a book was put up for sold less than 1 year or more than 1 year, cannot

identified tendency to sales but If book has been put up for sold less than 1 year and categories code have more than 21, have the opportunity to predicted correctly was 56% and tendency to sales in a “medium” class to 52.1%.

Table 4.12 Confusion matrix of 500 books from Store ID 83205.

83205	no	low	medium	high	veryhigh	sum
no	0	0	0	0	0	0
low	0	98	30	25	11	164
medium	0	56	43	13	12	124
high	0	45	20	36	20	121
veryhigh	0	31	16	20	24	91
sum	0	230	109	94	67	

According to Figures 4.1 – 4.3, it is evident that some of the stores are able to predict from the year the books were put on sale. However, some of the stores might have other primary factors than the year in making predictions. In some cases where it is difficult to make a prediction, i.e. the sales class cannot be predicted such as with Store ID 83205 *Year <=0* and *Cat_id <= 21* where the prediction accuracy is only 29%. If, however, the factor of price becomes involved, namely, *Price <= 90*, the prediction will be more accurate or will tend to be more precise.

According to the experiment, each of the stores is similar for the most part, i.e. the prediction depends upon the year. Some of the stores will depend upon the Cat_id which further relies upon the store type. Using a decision tree will help in greater awareness of the importance of each factor which will help explain in the first two methods where the differences are undeterminable, by using the train data. Furthermore, the consideration of individual stores enables the use of this prediction in considering the working style of a person’s own store by predicting and understands which particular patterns of attributes can determine the class more accurately with the model used such as Store ID 11218. If the *Year is <= 0* and the *Price is <= 80*, this model is highly predictable.

From results of the experiment by using artificial neural network and decision tree methods with numeric input data concluded that if data type of input data are numeric data, should be used decision tree method for predict tendency of sales quantity. Because of from the experiment, decision tree method can accurately predict rather than artificial neural network method. Addition, decision tree method has over precision and recall value. But this input data can be done if only Cat_id and Pub_id are organized neatly that the closer the value of Cat_id or Pub_id refer to similar categories or publisher.

4.4 Nominal data

Nominal data is indicating a separate category in the data that have no numerical meaning, cannot be calculated. Each value of the data will be presented in different input or output nodes.

4.4.1 Artificial Neural Network (ANN)

In this research, the ANN method uses a Multi-Layer Perceptron algorithm and hyperbolic tangent function as activation function. The hidden units were set to 40. Input node consisting of 243 nodes are Cat_id with 25 nodes, Pub_id with 216 nodes, Price with 1 node and Year with 1 node which data type of Cat_id and Pub_id are nominal data. The output node consisting of 5 nodes is the prediction value of sales quantity (Qty) in a form of different probabilities of 5 different classes. The expected outcome will be represented in the form of a confusion matrix to find precision, recall and accuracy as shown in Table 4.13

Table 4.13 Confusion matrix of the ANN method which data type of input data is nominal data.

		PREDICTED						
	ANN	no	low	medium	high	veryhigh	Recall (%)	sum
ACTUAL	no	153	65	45	40	49	43	352
	low	68	466	270	227	225	37	1256
	medium	53	285	386	213	191	34	1128
	high	49	254	243	315	275	28	1136
	veryhigh	26	223	169	197	513	45	1128
	Precision (%)	44	36	35	32	41	37.6 / 37.45	
	sum	349	1293	1113	992	1253		Average (%)
	Accuracy (%)							36.66%

Table 4.13 shows the confusion matrix of 5,000 books from 10 different stores by the ANN method, where a “high” class has the lowest precision and recall at 32% and 28%, respectively. The decision tree method could accurately predict the “high” class 315 books, while the books predicted in a “high” class at a total of 992 books and the actual value is in a “high” class at a total of 1136 books. On the other hand, the class that had the highest precision was a “no” class at 44%, decision tree method can accurately predict 153 books of the books predicted in this class at a total 349 books. And the highest recall is a “veryhigh” class at 45%, decision tree method can accurately predict 513 books of the actual value in this class at a total 1128 books.

The experimental results yielded by the decision tree method have an overall accuracy of 36.66%.

4.4.2 Decision Tree

The decision tree method was used with a J48 algorithm where the confidence factor was again set at 0.25. The input data consisted of the Cat_id, Pub_id, Price, and Year, which data type of Cat_id and Pub_id are nominal data. Target value is sales quantity (Qty), was converted to class form. Decision trees involving the creation of a tree to help decide based on the given inputs as well as numeric data. The expected outcome in terms of accuracy will be represented in a form confusion matrix to determine the precision, recall and accuracy values as shown in Table 4.14

Table 4.14 Confusion matrix of the decision tree method which data type of input data is nominal data.

		PREDICTED						
	Decision tree	no	low	medium	high	veryhigh	Recall (%)	sum
ACTUAL	no	169	89	38	19	37	48	352
	low	74	576	212	181	213	46	1256
	medium	64	350	331	194	189	29	1128
	high	54	278	206	343	255	30	1136
	veryhigh	28	216	101	173	610	54	1128
	Precision (%)	43	38	37	38	47	41.5 / 40.67	
	sum	389	1509	888	910	1304		Average (%)
	Accuracy (%)							40.58%

Table 4.14 shows the confusion matrix of 5,000 books from 10 different stores by the decision tree method, where a “veryhigh” class has the highest precision and recall at 47% and 54%, respectively. The decision tree method could accurately predict the “veryhigh” class 610 books, while the books predicted in a “veryhigh” class at a total of 1304 books and the actual value was in a “veryhigh” class at a total of 1128 books. On the other hand, the class that has the lowest precision and recall is a “medium” class, at 37% and 29% respectively. The decision tree method can accurately predict the “medium” class 331 books, while the books predicted in a “medium” class at a total of 888 books and the actual value is in a “medium” class at a total of 1128 books.

The experimental results yielded by the decision tree method have an overall accuracy of 40.58%. The results are displayed in form of decision tree as well as numeric data. But tree resulting from this data type is too large and too complex, to understand, making it difficult for analysis, and make decisions. Therefore, the representation form of conditions from decision tree that was made of the input data, by Table 4.15 is a part of example of conditions that obtained from a full decision tree which the full decision tree is shown in the Appendix C.

Table 4.15 Example of conditions that obtained from decision tree.

Root Node (L1)	L2	L3	L4	L5
Pub_id = 1				
	Year <= 0: veryhigh (11.0/9.0)			
	Year > 0			
		Year <= 1: medium (9.0/5.0)		
		Year > 1: low (2.0/2.0)		
Pub_id = 3: low (2.0/1.0)				
Pub_id = 5: medium (4.0/2.0)				
Pub_id = 10: high (1.0/1.0)				
Pub_id = 11: no (1.0/1.0)				
Pub_id = 13				
	Year <= 7: veryhigh (3.0/2.0)			
	Year > 7: low (5.0/3.0)			
Pub_id = 21				
	Year <= 0: veryhigh (4.0/4.0)			
	Year > 0: medium (4.0/3.0)			
Pub_id = 22: veryhigh (5.0/3.0)				
Pub_id = 23: medium (4.0/2.0)				
Pub_id = 24				
	Price <= 230: low (4.0/2.0)			
	Price > 230: high (3.0/2.0)			

Table 4.15 shows examples of conditions that obtained from decision tree of Store ID 10301. The orange color line represent node level 1 or the root node, the blue color line represent node level 2, the purple color line represent node level 3, the green color line represent node level 4 and the red color line represent node level 5. The numbers in parentheses consist of 2 values which are the number of the books predicted to that class and the number of books correctly predicted. Therefore this table can be described as follows: If *Pub_id = 1* cannot predict tendency of sales quantity, should be considered next level node that is node level 2 which is *Year <= 0* or *Year >0*. If *Pub_id = 1* and *Year <= 0* has trends to be sold books which look like this in a "veryhigh" class, has precision value at 81.81%. But if Pub_id = 1 and *Year > 0* cannot predict tendency of sales quantity, should be considered next level node that is node level 3 which is *Year <= 1* or *Year > 1*. If *Pub_id = 1* and *Year = 1* has trends to be sold books which look like this in a "medium" class, has precision value at 55.56%. But if *Pub_id = 1* and *Year > 1* has trends to be sold books which look like this in a "low" class, has precision value at 100%.

From results of the experiment by using artificial neural network and decision tree method with input data as nominal data concluded that if data type of input data are nominal data, the decision tree method should be used for predict tendency of sales quantity. Because of from the experiment, decision tree method can accurately predict rather than artificial neural network method. Addition, decision tree method has over precision and recall value. But decision tree obtained from the use of this data type is larger and more complex, making it difficult to used analysis, to help make decisions.

4.4.3 Collaborative Filtering

The CF method is another method that was used to predict tendency of sales quantity. The CF method calculates similarity by using the different characteristics and attributes of each book: categories code (Cat_id), publisher code (Pub_id), Price and Year. The nine books with the highest similarity to the active book, which are referred to as the neighboring books, have been selected. The attributes of the neighboring books are used to calculate the weight between the active books and each neighboring book. The prediction was then calculated by the weight average on the purchasing history of the selected neighboring books. The outcome prediction has been summarized in the form of accuracy, recall and precision as shown in Table 4.16.

Table 4.16 Confusion matrix of the CF method.

		PREDICTED						
	CF	no	low	medium	high	veryhigh	Recall (%)	sum
ACTUAL	no	10	162	77	54	49	3	352
	low	5	294	337	300	320	23	1256
	medium	3	212	419	260	234	37	1128
	high	3	116	375	313	329	28	1136
	veryhigh	0	49	234	310	535	47	1128
	Precision (%)	48	35	29	25	36	27.6 / 34.6	
	sum	21	833	1442	1237	1467		Average (%)
	Accuracy (%)							31.42

Table 4.16 shows the confusion matrix of 5,000 books from 10 different stores by the CF method. When the recall of the "no" class, is 0.03, or a prediction with confidence at only 3%. Because prediction of the "no" class is very small in this

study when compared to the entire set of data in the "no" class, the actual value or the number of books of the "no" class is 352, while the CF method can accurately predict only 10 of the aforementioned. On the other hand, the “veryhigh” class has a recall of 47%. The actual value is in a “veryhigh” class with a total of 1,128, while the CF method can accurately predict 535 of this total. If considered in terms of precision, however, the CF method has the precision of the “no” class at 48%. The books predicted in the “no” class at a total of 21, while the CF method can only accurately predict 10 of the aforementioned. On the other hand, the “high” class has the lowest precision at 25%. The books predicted in a “high” class with a total of 1,237, while the CF method can accurately predict 313 of the aforementioned. It may be concluded, therefore, that if there is a lot of data in the "no" class, the results may be more accurate with the CF method which has an overall accuracy of 31.42% according to the experiment results.

To give an example, 12 books from 3 bookstores, namely, Store ID 11218, 30121 and 83205, were used as input data for the analysis. The attributes are Cat_id, Pub_id, Price and Year. The actual quantity (Actual Qty), predict quantity (Predict Qty), Actual class and Predict class are also presented in Table 4.17 – 4.19.

Table 4.17 Examples of 4 books of a Store ID 11218 with its 9 neighboring books.

Book ID	**Cat_id**	**Pub_id**	**Price**	**Year**	**Actual Qty**	**Actual Class**	**Predict Qty**	**Predict Class**
Active Book								
75	14	53	150	0	0	no	778	high
Neighboring books								
48	9	50	150	1	0	no		
226	9	111	150	1	2	low		
231	9	111	150	1	24	veryhigh		
265	1	152	150	1	17	veryhigh		
274	1	89	150	1	1	low		
311	21	92	150	1	3	medium		
319	12	201	150	1	8	veryhigh		
489	8	107	150	1	3	medium		
103	9	25	155	1	12	veryhigh		

Table 4.17 Examples of 4 books of a Store ID 11218 with its 9 neighboring books. (Continued)

Book ID	Cat_id	Pub_id	Price	Year	Actual Qty	Actual Class	Predict Qty	Predict Class
Active Book								
162	1	47	100	1	10	veryhigh	10	veryhigh
Neighboring books								
158	1	47	165	1	6	high		
166	1	47	600	1	5	high		
146	2	66	98	2	4	medium		
310	21	92	110	2	12	veryhigh		
352	10	111	220	3	7	high		
271	11	22	90	2	15	veryhigh		
364	11	152	90	2	26	veryhigh		
437	11	72	350	5	1	low		
465	2	48	260	4	14	veryhigh		
Active Book								
387	9	111	100	9	3	medium	7.222	high
Neighboring books								
306	2	13	90	9	7	high		
379	1	47	65	7	4	medium		
390	23	125	65	7	1	low		
428	1	47	60	6	11	veryhigh		
297	21	13	100	9	7	high		
389	9	111	100	8	16	veryhigh		
307	2	13	90	8	9	veryhigh		
331	1	267	60	7	2	low		
427	1	47	60	7	5	high		

Table 4.17 Examples of 4 books of a Store ID 11218 with its 9 neighboring books. (Continued)

Book ID	Cat_id	Pub_id	Price	Year	Actual Qty	Actual Class	Predict Qty	Predict Class
Active Book								
391	23	125	65	6	6	high	5.667	high
Neighboring books								
379	1	47	65	7	4	medium		
306	2	13	90	9	7	high		
428	1	47	60	6	11	veryhigh		
331	1	267	60	7	2	low		
427	1	47	60	7	5	high		
447	1	47	30	4	3	medium		
297	21	13	100	9	7	high		
387	9	111	100	9	3	medium		
307	2	13	90	8	9	veryhigh		

Examples of 4 books of a Store ID 11218 with its 9 neighboring books with their purchasing history and classes. The overall accuracy using the CF method for this store is 25%

The table also shows the details of the attributes of each book, used as input data for the analysis of Cat_id, Pub_id, Price and Year. The actual quantity, predict quantity, actual class and predict class.

The prediction of Store ID 11218 for Book ID 162 was accurate with actual and predicted classes rated as “veryhigh” from averaging its 9 neighboring books. If we carefully investigate each neighboring book, the books still range mostly in the high and veryhigh classes, given the average of the veryhigh class. Therefore, it can be concluded that, with this particular set of attributes, sufficient information can be provided that this book is more’ likely’ to yield at least a high purchase.

When considering Book ID 75, the actual class is “no” while the predicted class is high. This means that the CF method predicted incorrectly. According to the table, the neighboring books of this Book ID come from nearly all possible classes and can indicate the poor quality of its attributes in representing the outcome of the class. The book with similar attributes can barely tell the correct potential of this book purchase. If different numbers of neighboring books are to be chosen, this book will obviously yield a different prediction result.

Some sets of attributes are insufficiently representative to identify the outcome prediction of the books. Therefore, with similar attributes, the actual purchasing quantity can be distributed in many different classes. While some sets of attributes can give some clues and can correctly estimate the class into which the purchasing quality of the falls.

Table 4.18 Examples of 4 books of a Store ID 30121 with its 9 neighboring books.

Book ID	Cat_id	Pub_id	Price	Year	Actual Qty	Actual Class	Predict Qty	Predict Class
Active Book								
98	11	94	140	7	3	medium	3.706	medium
Neighboring books								
100	11	94	140	8	8	veryhigh		
99	11	94	140	6	3	medium		
76	11	94	159	6	11	veryhigh		
75	11	94	159	5	1	low		
381	10	250	890	0	1	low		
380	10	250	890	1	0	no		
407	2	87	650	0	3	medium		
147	8	87	600	0	1	low		
408	2	87	650	1	4	medium		
Active Book								
118	1	87	50	0	0	no	4.895	medium
Neighboring books								
17	1	87	225	1	4	medium		
4	1	87	100	0	1	low		
5	1	87	100	1	12	veryhigh		
150	1	87	395	1	5	high		
18	1	87	225	0	10	veryhigh		
119	1	87	50	1	5	high		
149	1	87	395	0	2	low		
339	2	86	80	9	2	low		
95	2	144	125	9	3	medium		

Table 4.18 Examples of 4 books of a Store ID 30121 with its 9 neighboring books. (Continued)

Book ID	Cat_id	Pub_id	Price	Year	Actual Qty	Actual Class	Predict Qty	Predict Class
Active Book								
318	24	64	260	0	0	no	0.444	no
Neighboring books								
136	24	64	260	0	0	no		
323	24	64	260	0	0	no		
299	24	64	255	0	1	low		
355	24	64	265	0	0	no		
356	24	64	250	0	0	no		
138	24	64	245	0	0	no		
316	24	64	245	0	3	medium		
351	24	64	275	0	0	no		
357	24	64	275	0	0	no		
Active Book								
472	11	21	210	1	11	veryhigh	5.333	high
Neighboring books								
480	2	70	96	1	8	veryhigh		
30	2	19	95	1	2	low		
162	31	66	95	1	7	high		
5	1	87	100	1	12	veryhigh		
63	21	92	100	1	12	veryhigh		
170	1	47	100	1	5	high		
240	1	67	100	1	2	low		
361	4	61	100	1	0	no		
362	1	47	100	1	0	no		

Examples of 4 books of a Store ID 30121 with its 9 neighboring books with their purchasing history and classes. The overall accuracy for this store using the CF method is 24%.

The table also shows the details of the attributes of each book used as input data for the analysis of the Cat_id, Pub_id, Price and Year with the actual quantity, predict quantity, actual class and predict class.

The prediction of Store ID 30121 for Book ID 98 is accurate with the actual and predicted classes at “medium” from averaging its 9 neighboring books. According to the table, the neighboring books of this Book ID come from nearly all possible classes. Therefore, averaging the quality values of all the attributes yields a correct prediction mainly within the “medium” class, which is the same as Book ID 118. The neighboring books do not have a good form of attributes and cannot be clearly predicted. On the average, have the prediction will yield a “medium” class, but the actual class of this Book ID is not class the CF method predicted incorrectly. The CF method will predict class “no” because of the unique qualities of such attributes can be clearly predicted. In other words, nearly all of neighboring books must be in the “no” class, e.g. Book ID 318.

For Book ID 472, the actual class is veryhigh while the predicted class is high. Hence the CF method predicted incorrectly. According to the table, the neighboring books chosen have the form of attributes is clear up a little bit but predicted incorrectly which derives from the neighboring books were chosen.

Table 4.19 Examples of 4 books of a Store ID 83205 with its 9 neighboring books.

Book ID	**Cat_id**	**Pub_id**	**Price**	**Year**	**Actual Qty**	**Actual Class**	**Predict Qty**	**Predict Class**
Active Book								
22	24	64	245	2	1	low	1.555	low
Neighboring books								
25	24	64	245	2	2	low		
187	24	64	255	2	1	low		
189	24	64	255	2	1	low		
451	24	64	205	2	1	low		
322	24	64	115	1	1	low		
309	24	64	145	1	1	low		
486	24	64	160	1	4	medium		
332	24	64	190	1	2	low		
196	24	64	198	1	1	low		

Table 4.19 Examples of 4 books of a Store ID 83205 with its 9 neighboring books. (Continued)

Book ID	Cat_id	Pub_id	Price	Year	Actual Qty	Actual Class	Predict Qty	Predict Class
Active Book								
80	9	14	170	1	2	low	5	high
Neighboring books								
453	24	64	290	0	1	low		
31	24	64	295	0	9	veryhigh		
186	24	64	295	0	7	high		
495	24	64	295	0	3	medium		
228	6	25	285	0	5	high		
329	24	64	298	0	4	medium		
318	11	45	300	0	4	medium		
60	8	3	280	0	7	high		
168	2	82	280	0	5	high		
Active Book								
97	11	54	60	0	16	veryhigh	15.778	veryhigh
Neighboring books								
94	11	54	60	0	18	veryhigh		
98	11	54	60	0	11	veryhigh		
105	11	54	60	0	23	veryhigh		
108	11	54	60	0	16	veryhigh		
259	11	54	60	0	6	high		
93	11	54	40	0	15	veryhigh		
100	11	54	40	0	15	veryhigh		
102	11	54	40	0	28	veryhigh		
112	11	54	40	0	10	veryhigh		

Table 4.19 Examples of 4 books of a Store ID 83205 with its 9 neighboring books. (Continued)

Book ID	Cat_id	Pub_id	Price	Year	Actual Qty	Actual Class	Predict Qty	Predict Class
Active Book								
368	2	92	140	0	3	medium	7.713	high
Neighboring books								
383	9	38	490	10	1	low		
367	2	92	140	1	2	low		
146	0	49	12	9	4	medium		
116	1	267	60	9	3	medium		
269	1	86	200	9	5	high		
119	1	142	250	9	4	medium		
385	9	38	490	9	3	medium		
147	0	49	12	8	33	veryhigh		
118	1	267	60	8	15	veryhigh		

Examples of 4 books of a Store ID 83205 with its 9 neighboring books with their purchasing history and classes. The overall accuracy for this store using the CF method is 33%.

The table also shows the details of the attributes of each book used as input data for the analysis of the Cat_id, Pub_id, Price and Year, as well as the actual quantity, predict quantity, actual class and predict class.

The prediction of Store ID 83205 for Book ID 22 is accurate with the actual and predicted classes at “low”. This means that the CF method predicted correctly. Because the neighboring books were chosen, the form of attributes is clear.

For Book ID 80, the actual class is low while the predicted class is high. This means that the CF method predicted incorrectly. According to the table, the neighboring books of this Book ID do not have a good form of attributes and the outcome quantities are distributed to several classes.

According to Tables 4.17, 4.18 and 4.19, it can be concluded that there are 4 patterns of prediction as follows:

1. A correct prediction can be made because of the unique quality of such attributes and the aforementioned attributes can clearly indicate the correct class into

which the book would fall, e.g. Store ID 30121, Book ID 318; Store ID 83205, Book ID 22 and Store ID 83205, Book ID 97.

2. An accidental correct prediction is made due to the averaging of various forms of classes with the same attributes. Therefore, averaging the quality values of all of the attributes yields a correct prediction, mainly within the “medium” class such as Store ID 11218 Book ID 162, Store ID 11218 Book ID 391, and Store ID 30121 Book ID 98.

3. Incorrect predictions occur because the neighboring books do not have a good form of attributes and the outcome quantities are distributed to several classes. Hence, the prediction cannot be clearly made. On the average, the prediction will give the prediction as a “medium” class. Therefore, if the book is not in the medium class, it will be more likely to be incorrectly predicted such as Store ID 11218, Book ID 75; Store ID 11218, Book ID 387; Store ID 30121, Book ID 118; Store ID 83205, Book ID 80.

4. Incorrect predictions can also occur because neighboring books were chosen when the form of attributes is clearly slightly elevated, but predicted incorrectly which is derived from the neighboring books chosen such as Store ID 30121, Book ID 472 and Store ID 83205, Book ID 368.

Some books have different sale quantity from the other books with the same attributes and each book has its own unique sales quality without considering the pattern of the attributes, while the CF cannot yield a different outcome. Therefore, predicting the output for these particular types of books is difficult since many external factors are concerned.

Books with “no” or “low” classes will be difficult to predict since the calculation has a weighted average from the neighboring books. The "no" class will be predicted only when all of its neighbors are in the "no" class. Unless the pattern is unique enough to correctly predict. Furthermore, the selected attributes can mostly yield an initial prediction of each book; therefore, it is very hard for all neighboring books to be in the "no" class.

The CF method cannot retrieve a good recall for classes in the "no" class and can be more easily biased toward the classes with a higher amount of data due to the uncertain pattern of each particular class.

Additionally, this technique does not apply the important weight on each attribute; therefore it is likely that using all attributes equally can lead to an improper selection of the neighbor. As a result, the class with the smallest entity, such as the “no” class in this example, is less likely to be selected.

In addition, according to Table 4.16, it can be observed that the CF method will predict a “no” class less frequently than the other classes, but have the highest precision at 48% because the CF method can predict 10 books out of 21 books in the “no” class, which correctly predicts half of all the predictions. The CF can predict a class "no" less frequently because the CF method will predict a class "no" correctly, the neighboring books chosen must have a clear form of attributes, but there is no distribution. In other words, nearly at all of neighboring books must be in the "no" class such as Store ID 30121, Book ID 318. On the other hand, the neighboring books chosen have an unclear form of attributes with distribution to several classes such as Store ID 30121, Book ID 118. Neighboring books were chosen with distribution in low, medium, high and veryhigh classes. Hence, the CF method predicted incorrectly and there is greater chance for predicting the "medium" class than other classes. If the value is less obvious or distributed into multiple classes, the prediction will be averaged. On the average, the value is in the middle, namely, the "medium" class.

4.5 Experimental dividing the sales quantity into 3 classes.

This research originally divided the sales quantity into 5 classes (Table 3.2) with the accuracy value yielded by each of the abovementioned methods. When this same data is used, but divided into 3 classes, the accuracy values shown in Tables 4.20 - 4.22 will be obtained. It is evident, therefore, that dividing the data into 3 classes yields high accuracy for all 3 methods as shown in Table 4.23.

Table 4.20 Confusion Matrix of the CF method, in case that divide the sales quantity into 3 classes.

		PREDICTED				
	CF	low	medium	high	Recall (%)	sum
ACTUAL	low	471	414	723	29.29	1256
	medium	215	419	494	37.15	1128
	high	168	609	1487	65.68	1128
	Precision (%)	55.15	29.06	54.99	46.40 / 44.04	
	sum	833	1442	1237		Average (%)
	Accuracy (%)					47.54

Table 4.20 shows the confusion matrix of 5,000 books from 10 different stores by the CF method where the sales quantity (Qty) is the value to predict which is divided into 3 classes and has an overall accuracy of 47.54%.

Table 4.21 Confusion Matrix of the ANN method, in case that divide the sales quantity into 3 classes.

		PREDICTED				
	ANN	low	medium	high	Recall (%)	sum
ACTUAL	low	573	354	681	35.63	1608
	medium	306	257	565	22.78	1128
	high	306	342	1616	71.38	2264
	Precision (%)	48.35	26.97	56.46	43.93 / 43.27	
	sum	1185	953	2862		Average (%)
	Accuracy (%)					48.92

Table 4.21 shows the confusion matrix of 5,000 books from 10 different stores by the ANN method where the sales quantity (Qty) is the value to predict which is divided into 3 classes and has an overall accuracy of 48.92%.

Table 4.22 Confusion Matrix of the decision tree method, in case that divide the sales quantity into 3 classes.

		PREDICTED				
	Decision tree	low	medium	high	Recall (%)	sum
ACTUAL	low	860	306	442	53.48	1608
	medium	380	368	380	32.62	1128
	high	487	398	1379	60.91	2264
	Precision (%)	49.80	34.33	62.65	49.01 / 48.93	
	sum	1727	1072	2201		Average (%)
	Accuracy (%)					52.14

Table 4.22 shows the confusion matrix of 5,000 books from 10 different stores by the decision tree method where the sales quantity (Qty) is the value to predict which is divided into 3 classes and has an overall accuracy of 52.14%.

Table 4.23 The Accuracy of the 3 Methods by Dividing the Sales Quantity (Qty) into 3 and 5 Classes.

Accuracy (%)	3 Classes	5 Classes
CF	47.54	31.42
ANN	48.92	30.38
Decision Tree	52.14	40.1

From table 4.23, shows the accuracy of the 3 methods by dividing the sales quantity (Qty) into 3 and 5 classes which can be seen that, dividing the data into 3 classes yields high accuracy for all 3 methods, possibly because dividing the data into 3 classes gives a wider range for each class. Hence, there is a chance that the prediction will also be more accurate. However, because a wider range yields less information with fewer details and less precision, which is different from dividing the data into 5 classes, there is a chance the prediction will be an incorrect class if the values yielded by some of the predictions are erroneous. However, it can give different idea of books more accurately.

CHAPTER V
CONCLUSION

The researcher conducted this study with the objective of finding the most suitable method for predicting the sales volume for each book which can be sold in each retail store by using retroactive data from three years ago on the sales history and book details comprising data on the book type, publisher, price and duration of sales in order to help in deciding about the most suitable number of books that should be ordered for stock. Using people for such a prediction might be difficult and require skill and knowledge accumulating through the years. Furthermore, human errors can result in overstocking or shortages of stocks. Hence, a method or instrument should be found for application in decision-making in order to minimize errors and save time.

This research, the existing book data was analyzed by 3 methods was selected comprising (1) Artificial neural network (ANN), (2) Decision Tree and (3) Collaborative Filtering (CF). Input data was divided to 2 types by difference of data type which comprising (1) Numeric data and (2) Nominal data. Book detail used to as input data comprising book categories (Cat_id), book publisher (Pub_id), book price (Price), and duration of book sales (Year). In case data type of input data as numeric data, data type of Cat_id, Pub_id, Price, and Year as numeric data all. In case data type of input data as nominal data, data type of Cat_id and Pub_id as nominal data. To analyzed the most appropriate method of this case study.

From the experimental to find importance of attribute by divide to 3 cases by difference of input data which comprising (1) Input data is only Cat_id, (2) Input data is only Pub_id, and (3) Input data are Pub_id and Cat_id. the accuracy values obtained from each case were 31.90%, 41.30%, and 44.68%, respectively. and from the results of this experiment concluded that attribute has the most importance is Pub_id because has the accuracy value rather than Cat_id, but if the both attributes was used input data together for prediction will obtained more accuracy because of some books not able to indicate trends sales quantity clearly by using only Pub_id, but if using together with Cat_id , would be can predict trends more clearly, made more accuracy.

According to the experiment that was conducted, the accuracy values obtained for ANN and Decision tree method, in case data type of input data as numeric data were 30.38% and 40.10%, respectively. And in case data type of input data as nominal data were 36.66% and 40.58%, respectively. In addition, has experiment by using collaborative filtering method, to analyze trends sales quantity from book that has similarity by using Cat_id, Pub_id, Price, and Year which data type of Cat_id and Pub_id as nominal data. to calculate the similarities value between book in the store. Which the accuracy values obtained for CF method is 31.42% that the accuracy not high may be because CF method is appropriate to finding the similarities of user or clustering of user rather than of product. So using this method is not fully effectively.

From two different input data types, the method was used to predict trends of sales quantity and has the most accuracy of the both data type is the decision tree method, which have the accuracy at 40.10% of input data that is numeric data and 40.58% of input data that is nominal data.

Furthermore, the decision tree method even enables an understanding of which book factors are important at each store because each store has similar book factors implemented in the analysis. The experiment conducted in Chapter IV, revealed how each of the factors were differently important in each store. For example, according to the results of the experiment in Chapter IV, “Year” was the most important factor for predictions for most stores whereby either old or new books can be viewed first, while Cat_id was cited as the most important factor in predictions for Store ID 30121. In other words, we would need to see whether the *Cat_id was ≥ 23*. Then we can look at the other factors, namely, year. According to the aforementioned examples, the important factors for each store become understandable by using the decision tree method. Apart from enabling an understanding of various factors which are important components to the sales of each store, the decision tree method can predict the sales results from the main components which can be found in each book, but it has disadvantages that the tree should be improved when having more information because of decision tree will be constructed based on training data.

When considering the confusion matrix obtained of the both data type, in Chapter 4, the results are rather not different such as when consider in term of the precision and recall, input data that is numeric data have the precision and recall at 40.4% and 41.2%, respectively. And input data that is nominal data have the precision and recall at 40.67% and 41.5%, respectively. But there is something clearly different, is decision tree. Decision tree was obtained of input data that is nominal data, will be larger and more complex than numeric data, making it difficult to used analysis, to help make decisions. And in case that input data is numeric data, Cat_id and Pub_id must be organized neatly that the closer the value of Cat_id or Pub_id refer to similar categories or publisher.

This research has used only the accuracy value for measure performance, which may be added other methods to use measuring performance together such as ground truth, like research example that was reviewing in Chapter 2, by use the experts or a person accustomed and related to this field directly, to predicted trends sales quantity, to compare with the results from each method were select. Especially in the CF method that often used the ground truth to measured performance. However, ground truth may take a much, if the data has quite high volume, which is why this research has not use ground truth.

From researching, researcher has the recommendations that, the accuracy value can be increased if using a more factors such as best seller books, awarded authors, awarded books, sequels and bookstore locations, etc. Because of the present has many factors that influence a buyer's decision to buy the books even more. Hence, the researcher would like to recommend that, if the factors applied to the analysis are increased, book sales predictions might be more accurate and precise.

Finally, from the accuracy was obtained, if compared to the decision by the people considered to have a sufficiently high accuracy, without the risk of human error, can assist in determining, reducing time and increasing accuracy, which may be used as the preliminary decision to people used to easier.

REFERENCES

1 Francesco Ricci, Lior Rokach, Bracha Shapira. Recommender systems handbook. NY: Springer; 2011.

2 Pentaho Community MultilayerPerceptron [homepage on the Internet]. 2012 [cited 2012 June 25]. Available from: http://wiki.pentaho.com/display/DATAMINING/MultilayerPerceptron

3 Prem Melville, Vikas Sindhwani. Recommender systems. Encyclopedia of Machine Learning. New York: Yorktown Heights; 2010.

4 Dan-Er Chen. The Collaborative Filtering Recommendation Algorithm Based on BP Neural Networks; International Symposium on Intelligent Ubiquitous Computing and Education; 2009.

5 Greg Linden, Brent Smith, Jeremy York. Amazon.com Recommendations Item-to-Item Collaborative Filtering, Amazon.com: IEEE Computer Society; 2003.

6 George Lekakos, George M Giaglis. Improving the prediction accuracy of recommendation algorithms: Approaches anchored on human factors. Interacting with Computers. 2006; 18:410–431.

7 Tae Hyup Roh, Kyong Joo Oh, Ingoo Han. The collaborative filtering recommendation based on SOM cluster-indexing CBR. Expert System with Application. 2003; 25:413-423.

8 Ya-Yueh Shih, Duen-Ren Liu. Product recommendation approaches: Collaborative filtering via customer lifetime value and customer demands. Expert Systems with Applications. 2007.

9 Hyung Jun Ahn. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. Information Sciences. 2008; 178:37-51.

10 John S Breese, David Heckerman, Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering; Proceedings of the Four-teenth

Conference on Uncertainty in Artificial Intelligence; 1998 July; Madison, WI.

11 Jonathan L Herlocker, Joseph A Konstan, Al Borchers, John Riedl. An Algorithmic Framework for Performing Collaborative Filtering; Proceedings of the 22nd International ACM SIGIR Conference on Research and Development in Information Retrieval; 1999.

12 C - H Lee, Y - H Kim, P - K Rhee. Web personalization expert with combining collaborative filtering and association rule mining technique. Expert System with Application. 2001; 21:131-137.

13 Neuro AI - Intelligent systems and Neural Networks [homepage on the Internet]. 2010 [cited 2012 June 25]. Available from: http://www.learnartificialneuralnetworks.com/

14 Neural networks [homepage on the Internet]. 2006 [updated 2006 Dec 16; cited 2012 June 25]. Available from: http://www-cs-faculty.stanford.edu/~eroberts/courses/soco/projects/2000-01/neural-networks/Architecture/feedforward.html

15 Lu WZ, Fan HY, Leung AY, Wong JC. Analysis of Pollutant Levels in Central Hong Kong applying Neural Network method with Particle Swarm Optimization. Environmental Monitoring and Assessment. 2002; 79:217–230.

16 Scribd.com [homepage on the Internet]. 2013 [cited 2013 August 10]. Available from: http://www.scribd.com/doc/95042193/MLP

17 Gardner MW, Dorling SR. Artificial neural networks the multilayer perceptron—a review of applications in the atmospheric sciences. Atmospheric Environment. 1998; 32:14–15.

18 Alsmadi MK, Omar KB, Noah SA, Almarashdah I. Performance comparison of multi-layer. Perceptron (Back Propagation, Delta Rule and Perceptron) algorithms in neural networks; Proceedings of the IEEE International Advance Computing Conference; 2009 March 6-7; 296-299.

19 Alsmadi MK, Omar KB, Noah SA, Almarashdah I, 2011. A Hybrid Memetic Algorithm with Back-propagation Classifier for Fish Classification Based

on Robust Features Extraction from PLGF and Shape Measurements. Information Technology Journal, 10: 944-954.

20 Panida Padungweang, Sirapat Chiewchanwattana, Khamron Sunat. Artificial Consultative Thai University Admission Based on MLP; Conference on Knowledge and Smart technologies; 2009.

21 Chutchai Keawta and Autchara Mahaweerawat. การวินิจฉัยคดีด้วยเทคนิคต้นไม้ตัดสินใจ. NCIT. 2010; 308 -313.

22 Quinlan JR. Introduction of Decision Trees. Machine Learning. 1986; 1(1):81-106.

23 Quinlan JR. C4.5: program for machine learning, California: Morgan Kaufmann; 1993.

24 Payam Refaeilzadeh, Lei Tang, Huan Liu: Cross-Validation. Encyclopedia of Database Systems. 2009; 532-538

25 Paranya Ammaruekarat, Phayung Meesad. A Comparative Efficiency of Dimensionality Reduction and Neural Network Classification; Proceedings of the 11th graduate research conference; 2010.

26 Hongsheng L, Yongcai Y. An Experiential Analysis of the Influence of Critical Marketing Environment on Consumer Decision Making; International Conference on E-Business and E-Government; 2010.

27 Ramı´rez M, Velho H, Ferreira N. Artificial neural network technique for rainfall forecasting applied to the Sa˜o Paulo region. Journal of Hydrology. 2005; 301:146 – 162.

28 Park SJ, Hwang CS, Vlek PLG. Comparison of adaptive techniques to predict crop yield response under varying soil and land management conditions. Agricultural Systems. 2005; 85:59–81.

29 Alon I, Qi M, Sadowski RJ. Forecasting aggregate retail sales: a comparison of artificial neural networks and traditional methods. Journal of Retailing and Consumer Services. 2001; 8:147-156.

30 Henry C Co, Rujirek Boosarawongse. Forecasting Thailand's rice export: Statistical techniques vs. artificial neural networks. Computers and Industrial Engineering. 2007 Nov; 53(4):610-627.

31 Chinna Srachoom. Application of Artificial Neural Network for Weather Forecast [MSc thesis]. Chiangmai: Chiang Mai University; 2550.

32 Tassanawan Kaewsai, Supoj Nitsuwat. Movies Recommender System using Collaborative Filtering and K-Means; Proceedings of the 5th National Conference on Computing and Information Technology (NCCIT); 2009.

33 Yugal kumar, G Sahoo. Analysis of Bayes, Neural Network and Tree Classifier of Classification Technique in Data Mining using WEKA. Computer Science & Information Technology (CS & IT); 2012.

34 Arihito Endo, Takeo Shibata, Hiroshi Tanaka. Comparison of Seven Algorithms to Predict Breast Cancer Survival. Biomedical Soft Computing and Human Sciences. 2008; 13(2):11-16.

35 Krit Somkantha, Wilaiporn Kultangwattana, Worrawit Kultangwattana. The Development of Personnel Assessment System by Artificial Neural Network. Information Technology Journal. 2013; 9:58-66.

APPENDICES

APPENDIX A
The details of Cat_id, Pub_id and Store_id

1. The details of Cat_id

Table A.1 Shows categories name (Cat_name) of each Cat_id that using in this research

Cat_id	Cat_name
0	เบ็ดเตล็ด/ปกิณกะ
02	ศิลปะและการวาดเขียน / วาดภาพ
1	ศาสนา / ปรัชญา / จิตวิทยา
011	ศาสนา / ปรัชญา
013	โหราศาสตร์
2	วรรณกรรม
21	วรรณกรรม / เรื่องสั้น / นิยาย
23	บทกวี / กวีนิพนธ์
24	นวนิยาย / โรมานซ์ / นิยายไทย
25	นิยายไทย
3	สารคดี / โบราณคดี
31	สารคดีท่องเที่ยว
4	วรรณกรรมเยาวชน
5	บันเทิง / หรรษา / ชวนหัว
6	วรรณกรรมแปล
7	การเมือง / เศรษฐกิจ / สังคม (วิจารณ์)
8	ประวัติศาสตร์ / วัฒนธรรม / อัตชีวประวัติ
9	บุคคล / ครอบครัว / บทบันทึก / จิตวิทยา
10	บริหารธุรกิจ / ธุรกิจศึกษา

Table A.1 Shows categories name (Cat_name) of each Cat_id that using in this research (Cont.)

Cat_id	Cat_name
11	ความรู้ทั่วไป / วิชาการ
111	ความรู้ทั่วไป / วิชาชีพ
112	วิชาการ / ภาษาศาสตร์
12	สุขภาพ / สมุนไพร
13	เกษตรกรรม
14	วารสาร / นิตยสาร

2. The details of Pub_id

Table A.2 Shows publisher name (Pub_name) of each Pub_id that using in this research

Pub_id	Pub_name
1	Best 4 kids
2	Ink (อิงค์)
3	VOTE (บ. ดาวเหนือ พับลิชชิ่ง จก.)
4	Way of book
5	สำนักพิมพ์เกียรติวรรณ อมาตยกุล, อ.
6	สำนักพิมพ์กระท่อม ปล.
7	สำนักพิมพ์กราฟิคอาร์ต
8	เก้าแต้ม
10	เขียน
11	สำนักพิมพ์คมบาง
12	สำนักพิมพ์คบไฟ
13	บริษัทเคล็ดไทย จำกัด
14	เคล็ดไทย/กาละแมร์

Table A.2 Shows publisher name (Pub_name) of each Pub_id that using in this research (Cont.)

Pub_id	**Pub_name**
15	เคล็ดไทย/สุเมธ
16	เคล็ดไทย/เฟอร์เมทสยาม
17	เคล็ดไทย/ฟ้าเดียวกัน
19	เคล็ดไทย/สำนักพิมพ์หนึ่ง
20	เคล็ดไทย/Snow book
21	เคล็ดไทย/โอเพ่น
22	เคล็ดไทย/บรรเทิง
23	เคล็ดไทย/สวนเงินมีมา
24	เคล็ดไทย/ฟรีฟอร์ม
25	สำนักพิมพ์คุณธรรม
26	ครอบครัว
27	เครือข่ายพุทธิกา
28	เคหวัตถุ
29	คริสตัล พับลิชชิ่ง (พาเพลิน)
30	คลองบางหลวง
31	เจ.บี.พับลิชชิ่ง
33	บริษัท ชุณหวัตร จำกัด
34	ชีส พับลิชชิ่ง
35	สำนักพิมพ์ชายขอบ
36	ชวนอ่าน
37	สำนักพิมพ์ดุจลดา
38	สำนักพิมพ์เดลฟี
39	บริษัท ดับบลิวโอเอฟเอส พับบลิชชิ่ง (ประเทศไทย) จำกัด
40	ดวงตะวัน
41	สำนักพิมพ์ต้นธรรม
42	สำนักพิมพ์ตรงหัว

Table A.2 Shows publisher name (Pub_name) of each Pub_id that using in this research (Cont.)

Pub_id	Pub_name
43	สำนักพิมพ์ต้นไม้
44	ตากับยายในพระจันทร์
45	ที่ปรึกษากฎหมาย เมริท จำกัด
46	ไทกอน
47	สำนักพิมพ์ธรรมดา
48	สำนักพิมพ์นาคร
49	สำนักพิมพ์บ้านมงคล/วิชญา
50	สำนักพิมพ์บันลือกิจ
51	บริษัท บิสซี่เดย์ จำกัด
52	เบญจา มังคละพฤกษ์
53	สำนักพิมพ์ปาจารยสาร
54	ปิกัสโซ
55	บริษัทโปรแอทติวา
56	ทันตแพทย์ ปริญญา ใจเย็น
57	บ. ประพันธ์สาส์น จก
58	ปิยศักดิ์ อู่ทรัพย์ (คุณ)
60	ผจญภัย (ศิริวร แก้วกาญจน์)
61	นายพีระ บุญจริง
62	พิมพ์บูรพา-สามัญชน
63	พาเพลิน
64	พิมพ์ใจ (กระซิบพับลิชชั่น)
65	ฟ้าเดียวกัน
66	ฟรีฟอร์ม
67	สำนักพิมพ์ภาษิต
68	สำนักพิมพ์มูลนิธิโกมลคีมทอง
69	สำนักพิมพ์เมน'สคลับ (บ้านหนังสือ)

Table A.2 Shows publisher name (Pub_name) of each Pub_id that using in this research (Cont.)

Pub_id	**Pub_name**
70	สำนักพิมพ์ไม้ยมก
71	มูลนิธิเพื่อผู้บริโภค
72	สำนักพิมพ์มายิก (เดลพี)
73	เม่นวรรณกรรม
74	มนวิภา อาวิพันธุ์ (คุณ)
75	ระฆังทอง
76	รุ่งโรจน์ สุวรรณธาดา
77	ร้านเล่า
78	ลูกหมาตาดำๆ
79	เลิฟ ออฟ ดรีมเมอร์
80	สำนักพิมพ์ไว้ลาย
81	สำนักพิมพ์วิภาษา
82	วสี ครีเอชั่น
83	บริษัท วีเลิร์น จำกัด
84	วิทยาลัยดุริยางคศิลป์ มหาวิทยาลัยมหิดล
85	ศูนย์ไทย-ธิเบต
86	สำนักพิมพ์ศยาม
87	สำนักพิมพ์ศรีปัญญา
88	ศูนย์มานุษยวิทยาสิรินธร (องค์การมหาชน)
89	ศูนย์ศึกษาและพัฒนาสันติวิธี มหาวิทยาลัยมหิดล
91	สายส่งศึกษิต
92	สายส่งศึกษิต/เคล็ดไทย
93	สายส่งศึกษิต/โอเพ่น
94	สายส่งศึกษิต/ประเพศ
95	สายส่งศึกษิต/บรรเทิง
96	สายส่งศึกษิต/กุดจี่

Table A.2 Shows publisher name (Pub_name) of each Pub_id that using in this research (Cont.)

Pub_id	**Pub_name**
98	สำนักพิมพ์สามัญชน
99	นายสำรวย นักการเรียน
100	สวนเงินมีมา
101	สนามกอล์ฟ
102	สมมติ
103	สยามมิส พับลิชชิ่ง เฮ้าส์
104	สายใยประชาธรรม
105	บริษัท สมใจบุ๊คส์ จำกัด
106	เสฐียรพงษ์ วรรณปก (อาจารย์)
107	สยามปริทัศน์
108	ทันตแพทย์เสรี ปาการเสรี
110	สำนักพิมพ์หอน
111	สำนักพิมพ์โหราคาร
112	หมูเพนกวิน
113	ห.จ.ก. กิมจั๊วพาณิชย์
114	ห้างหุ้นส่วนสามัญสำนักพิมพ์คุณธรรม
115	นายอำนาจ เจริญศิลป์
116	สำนักพิมพ์เอส.เจ.บุ๊คส์
117	คุณเอกชัย จุละจาริตต์
118	โอเพ่นบุ๊คส์
119	โอ้ พระเจ้าพับลิชชิ่ง
120	อินทภาษ
121	สำนักพิมพ์หนึ่ง
122	Shine publishing house
123	สายส่งศึกษิต/สามัญชน
124	เคล็ดไทย/สามัญชน

Table A.2 Shows publisher name (Pub_name) of each Pub_id that using in this research (Cont.)

Pub_id	**Pub_name**
125	สายส่งศึกษิต/รูปจันทร์
126	เคล็ดไทย/สำนักพิมพ์๑๐๐๑ ราตรี
127	เคล็ดไทย/รูปจันทร์
130	บริษัทมายด์ พับลิสชิ่ง
131	บริษัท สำนักพิมพ์สุวรรณภูมิ จำกัด
132	มายโรส (บ. บาลานซ์ พับลิชชิ่ง จก)
133	จอยบุ๊คคลับ
134	Venus Plus
135	บลูไดมอนด์ 2010 พับลิชชิ่ง
136	มูลนิธิอันวีกษณา
137	สนพ. คเณศบุรี จำกัด
138	มนตรี มุงคุณ (คุณ)
141	นานนะภงค์
142	สำนักพิมพ์คณะสังคมผาสุก
144	สำนักพิมพ์โป๊ยเซียน
146	บรรณาธิการ
147	พระครูสมุห์โพธิ์ จนฺทสีโล
150	มูลนิธิพุทธธรรม
151	สำนักพิมพ์อินทรีย์
152	น.พ.คงศักดิ์ ตันไพจิตร
153	สำนักพิมพ์สถาบันธรรมาธิปไตย
155	สำนักพิมพ์กำแก้ว
157	บริษัท ไบรทคิดส์ จำกัด (ไลต์เฮาส์พับลิชชิ่ง)
158	บริษัทศึกษิตสยาม จำกัด
159	ลานา 2553
160	วิศัลยา

Table A.2 Shows publisher name (Pub_name) of each Pub_id that using in this research (Cont.)

Pub_id	Pub_name
161	อรอุมา งามจิตพาณิชย์
162	สถาบันเศรษฐกิจพอเพียง
164	Paega Publishing
169	วรพจน์ พันธุ์พงศ์ (คุณ)
170	มูลนิธิชีววิถี (Biothai)
171	ลายสือ
172	เสมอขวัญ
173	ศูนย์เฝ้าระวังเชิงองค์ความรู้สถานการณ์ภาคใต้
174	สำนักพิมพ์พิราบ
176	โอเอซิส
177	วันทิพย์
178	ทวีสาส์น
181	ดร.บูฆอรี ยีหมะ
183	ออไร้เอิน
184	ลูก-หลาน จ่าง แซ่ตั้ง
185	สามสหาย
186	สำนักงานกฎหมายอัมสเตอร์ดัม แอนด์ เปรอฟ
187	ธนภูมิ บุญช่วยมั่นคง (คุณ)
189	ชุมศักดิ์ นรารัตน์วงศ์ (คุณ)
190	ศูนย์เรียนรู้ทุ่งสักอาศรม
191	วิภาวี
192	สถาบันสุวรรณภูมิอภิวัฒน์
193	เคล็ดไทย/สายส่งศึกษิต
194	เคล็ดไทย/ประเพศ
195	สภาพระธรรมกถึกแห่งประเทศไทย
196	เคล็ดไทย/มายด์ พับลิชชิ่ง

Table A.2 Shows publisher name (Pub_name) of each Pub_id that using in this research (Cont.)

Pub_id	Pub_name
197	จิตติมณฑน์ มหาวัจน์ (คุณ)
198	ก๊วนปาร์ตี้
201	มูลนิธิสายใยแผ่นดิน
204	พระมหาถนัด อตถจารี
206	สำนักพิมพ์คลาสสิก
210	เคล็ดไทย/ลูกหมาตาดำ ๆ
211	อิราโต้ พับลิชชิ่ง
212	สุธิดา อรรถภาษ
214	ประสาธน์ เกียรติไพบูลย์กิจ
215	ฟุตเวิร์ค
216	อภิชา ภาอารยพัฒน์ (รศ.)
218	ภูมิพัฒน์ พงษ์คำพรรณ (คุณ)
219	มูลนิธิพุทธมรดก
220	บริษัท ซิตี้เจอร์นัล มัลติมีเดีย
222	พระครูกัลยาณสิทธิ์วัฒน์
223	บริษัท เดอะเกรทไฟน์อาร์ท จำกัด
224	นายผศ . สานนท์ เจริญฉาย
225	สหกรณ์เครดิตยูเนี่ยนชุมชนคลองจั่น จำกัด
226	สำนักพิมพ์กอไผ่
228	สิริมงคลคำ (เสี้ยวจันทร์)
229	โฆษิต
231	บริษัท เพื่อนคิด จำกัด
233	อุทัย บุญเย็น (คุณ)
238	ห้างหุ้นส่วนจำกัด คอซาครีเอทีฟ
239	สุเมธ ณรงคฤทธิ์ (ไซม่อน)
244	วัลลภ ปัญญาชวนะ (คุณ)

Table A.2 Shows publisher name (Pub_name) of each Pub_id that using in this research (Cont.)

Pub_id	**Pub_name**
245	ชาตรี ศรีสารสกุล (คุณ)
246	หนึ่งเดียว
247	เคล็ดไทย/ใบไม้ป่า
250	บริษัท เทน ครีเอชั่น จำกัด
251	ป่งใบ
253	ลมหายใจ
255	พุงกาง
256	ฟ้าหลังฝน
259	ก้านดิน มีเดียบุ๊ค (venus plus)
262	ใต้แผ่นดินศยาม
263	บ้านนางฟ้า
266	โรงพิมพ์เลี่ยงเชียงจงเจริญ
267	พระยุทธพงษ์ ยุทธวังโส
272	เครือข่ายปฏิรูปที่ดินแห่งประเทศไทย
273	เคล็ดไทย/อินเตอร์สปาฯ
284	สำนักพิมพ์เจิดจ้า
285	ใบตอง
288	วงศ์สวัสดิ์
293	แมวบ้าน
294	เคล็ดไทย/สมาคมนักเขียนแห่งประเทศไทย
297	มูลนิธิผสานวัฒนธรรม

3. The details of Store_id

Table A.3 Shows store name (Store_name) of each Store_id that using in this research

Store_id	Store_name
10301	แผงบุ๊คเชส์ท
10725	บ. ซีเอ็ดยูเคชั่น จก.(มหาชน)หน่วยงานกลาง กทม.
11218	เคล็ดไทยขายงาน
20123	บริษัท แม็กซ์ วิชั่น จำกัด เซ็นทรัลรามอินทรา
30121	ศูนย์หนังสือสื่อภาษา เมืองทอง
50434	บ.สยามอินเตอร์มัลติมีเดีย จก.(มหาชน) สำนักงานใหญ่
63420	ท็อปแลนด์พลาซ่า พิษณุโลก
83205	บริษัท ประณอมทวี จำกัด จ.ชลบุรี
93301	บริษัท คลังพลาซ่า จำกัด
93302	บริษัท คลังพลาซ่า จอมสุรางค์ จำกัด จ.นครราชสีมา

APPENDIX B
MLP NEURAL NETWORK SIMULATION USING NEUROSOLUTIONS TOOLBOX FOR MATLAB

The process to train and test a designed MLP neural network:

1. We make training patterns and test patterns.

traindata1 = [cat1_tr; group1_tr; price1_tr; pub1_tr]
testdata1 = [cat1_te; group1_te; price1_te; pub1_te]

traindata1 is input data of training data series 1 that consist of categories code (cat1_tr), group code (group1_tr), price (price1_tr) and publisher code (pub1_tr).Testdata1 is input data of testing data series 1 that consist of categories code (cat1_te), group code (group1_te), price (price1_te) and publisher code (pub1_te). This research has 10 series because input data will be divided into training data and testing data by k-fold cross validation method which k is 10.

2. Network architecture should be defined by *newff* MATLAB function with the number of layers, neurons and transfer functions. For example in Figure 1, show create network. The following command can be used.

net1 = newff (traindata1, qty1_tr, 5)

traindata1 is input data of training data series 1, qty1_tr is order quantity data of this input data, 5 is number of hidden layer.

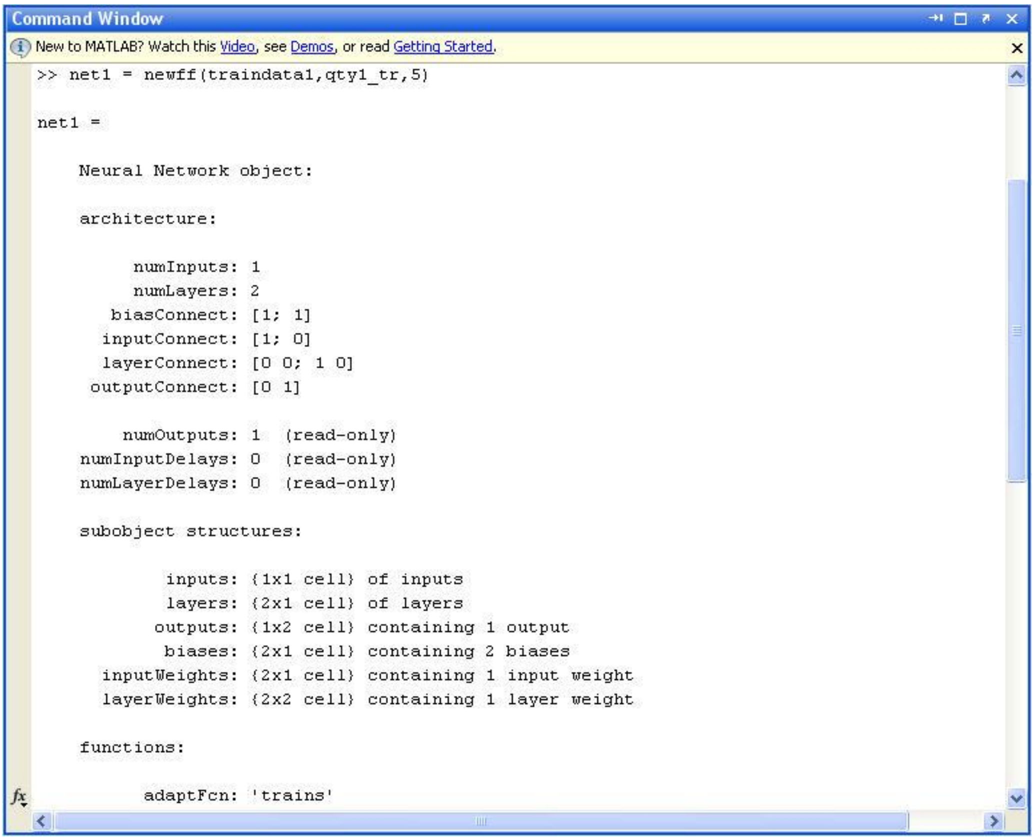


Figure B.1 Example of create network (more explanation).

3. The defined neural network architecture trained by *train* MATLAB function with input patterns and training parameters. For example in Figure.2 show training neural network. The following command can be used:

net1 = train (net1, traindata1, qty1_tr)

traindata1 is input data of training data series 1, qty1_tr is order quantity data of this input data, net1 is neural network architecture, was created in step 2.

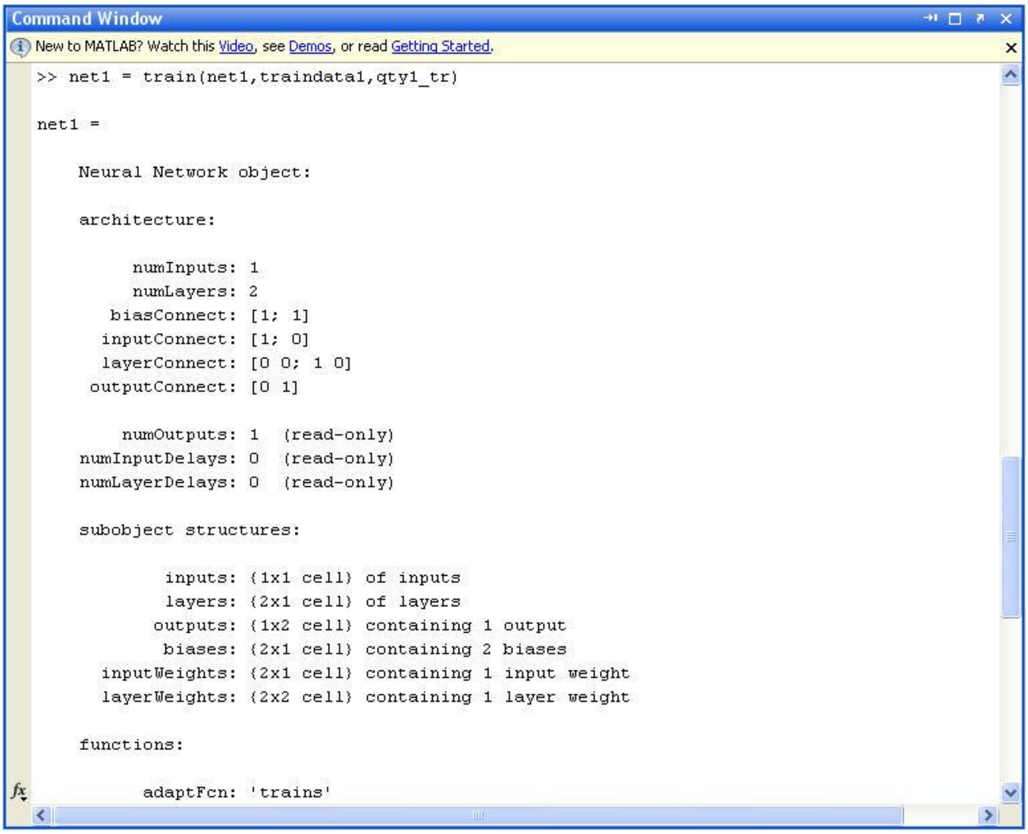
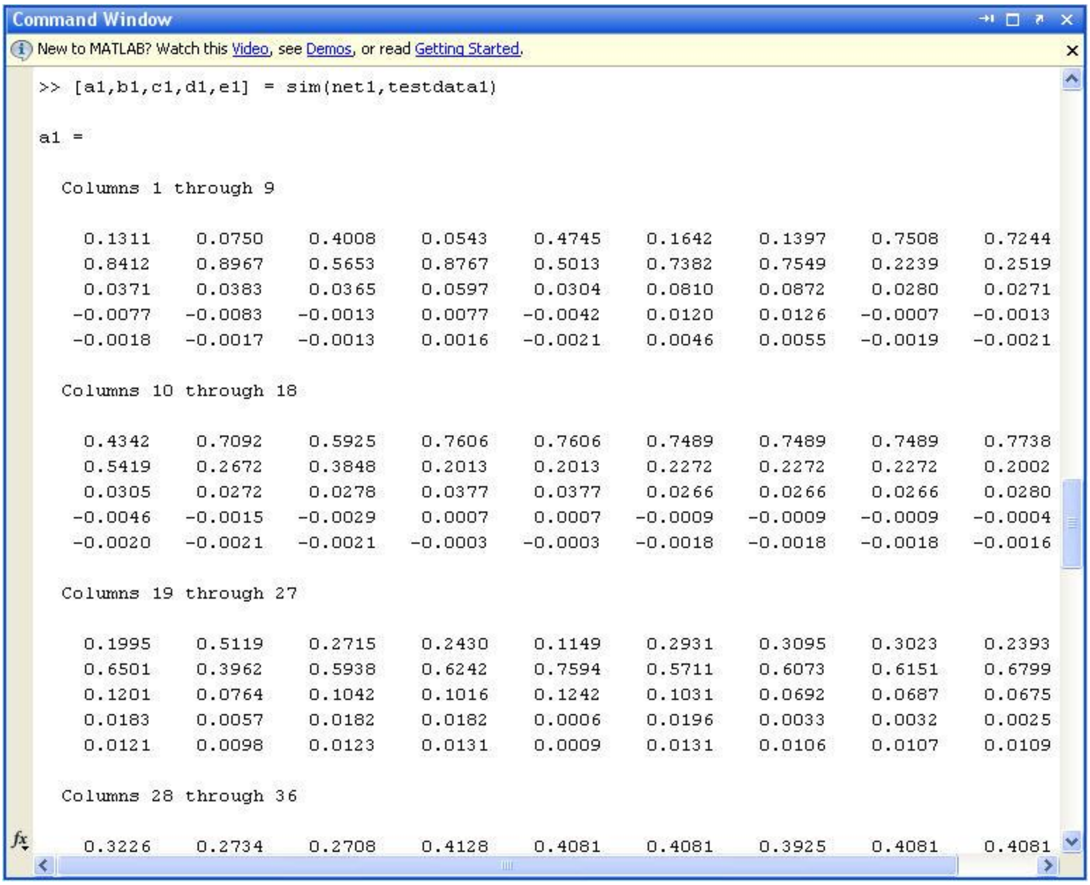


Figure B.2 Example of training.

4. We can easily check the result by using a *sim* MATLAB function. For example in Figure.3 show of testing from neural network that was created in process 2 and 3. The following command can be used.

[a1, b1, c1, d1, e1] = sim (net1, testdata1)

testdata1 is input data of testing data series 1, net1 is neural network architecture, was created in step 2 and training in step 3. a1, b1, c1, d1, e1 is result from using a *sim* MATLAB function. Which a1 are Network outputs, b1 are Final input delay conditions, c1 are Final layer delay conditions, d1 are Network errors, and e1 is Network performance. b1, c1 are optional and need only be used for networks that have input or layer delays.


Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
>> [a1,b1,c1,d1,e1] = sim(net1,testdata1)

a1 =

  Columns 1 through 9

    0.1311    0.0750    0.4008    0.0543    0.4745    0.1642    0.1397    0.7508    0.7244
    0.8412    0.8967    0.5653    0.8767    0.5013    0.7382    0.7549    0.2239    0.2519
    0.0371    0.0383    0.0365    0.0597    0.0304    0.0810    0.0872    0.0280    0.0271
   -0.0077   -0.0083   -0.0013    0.0077   -0.0042    0.0120    0.0126   -0.0007   -0.0013
   -0.0018   -0.0017   -0.0013    0.0016   -0.0021    0.0046    0.0055   -0.0019   -0.0021

  Columns 10 through 18

    0.4342    0.7092    0.5925    0.7606    0.7606    0.7489    0.7489    0.7489    0.7738
    0.5419    0.2672    0.3848    0.2013    0.2013    0.2272    0.2272    0.2272    0.2002
    0.0305    0.0272    0.0278    0.0377    0.0377    0.0266    0.0266    0.0266    0.0280
   -0.0046   -0.0015   -0.0029    0.0007    0.0007   -0.0009   -0.0009   -0.0009   -0.0004
   -0.0020   -0.0021   -0.0021   -0.0003   -0.0003   -0.0018   -0.0018   -0.0018   -0.0016

  Columns 19 through 27

    0.1995    0.5119    0.2715    0.2430    0.1149    0.2931    0.3095    0.3023    0.2393
    0.6501    0.3962    0.5938    0.6242    0.7594    0.5711    0.6073    0.6151    0.6799
    0.1201    0.0764    0.1042    0.1016    0.1242    0.1031    0.0692    0.0687    0.0675
    0.0183    0.0057    0.0182    0.0182    0.0006    0.0196    0.0033    0.0032    0.0025
    0.0121    0.0098    0.0123    0.0131    0.0009    0.0131    0.0106    0.0107    0.0109

  Columns 28 through 36

    0.3226    0.2734    0.2708    0.4128    0.4081    0.4081    0.3925    0.4081    0.4081
```

Figure B.3 Example of testing.

APPENDIX C
EXAMPLE OF DECISION TREE

1. Data Type is Numeric Data

1.1. Decision Tree of Store ID 11218

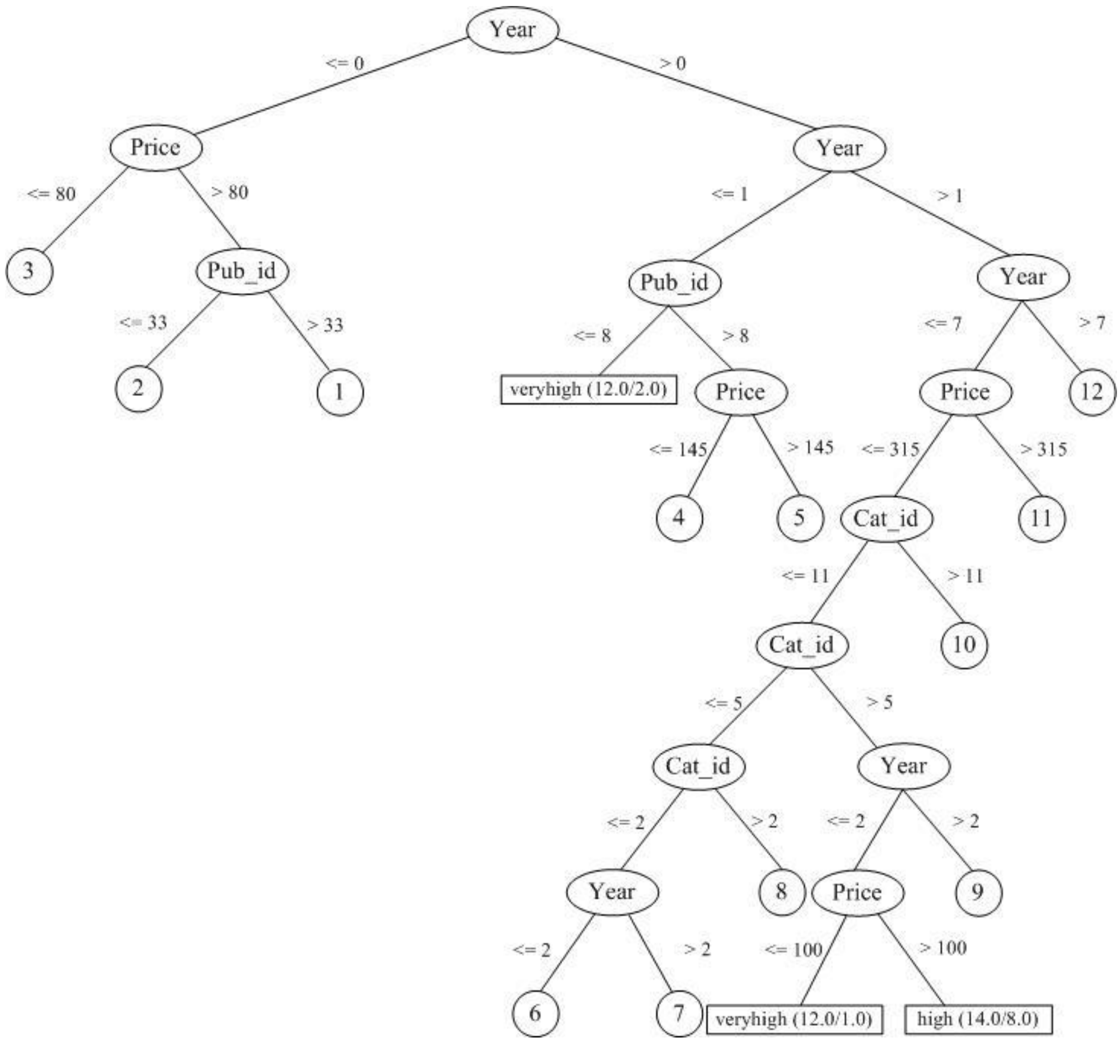


Figure C.1 Decision Tree of Store ID 11218.

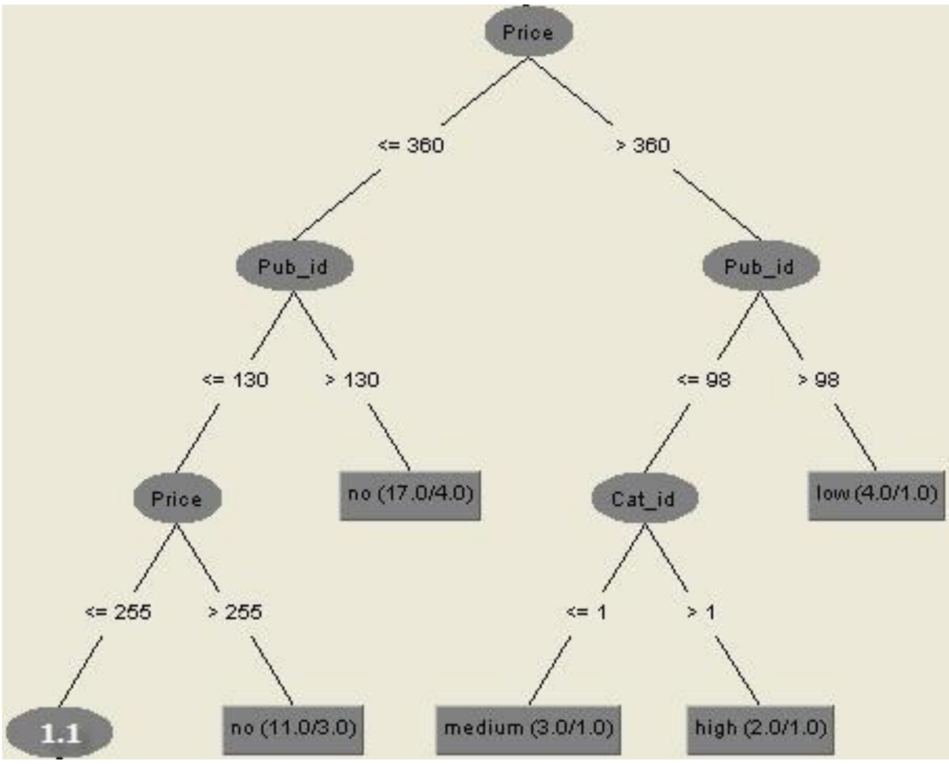


Figure C.2 Shown subtree that represent node 1 in Figure C.1.

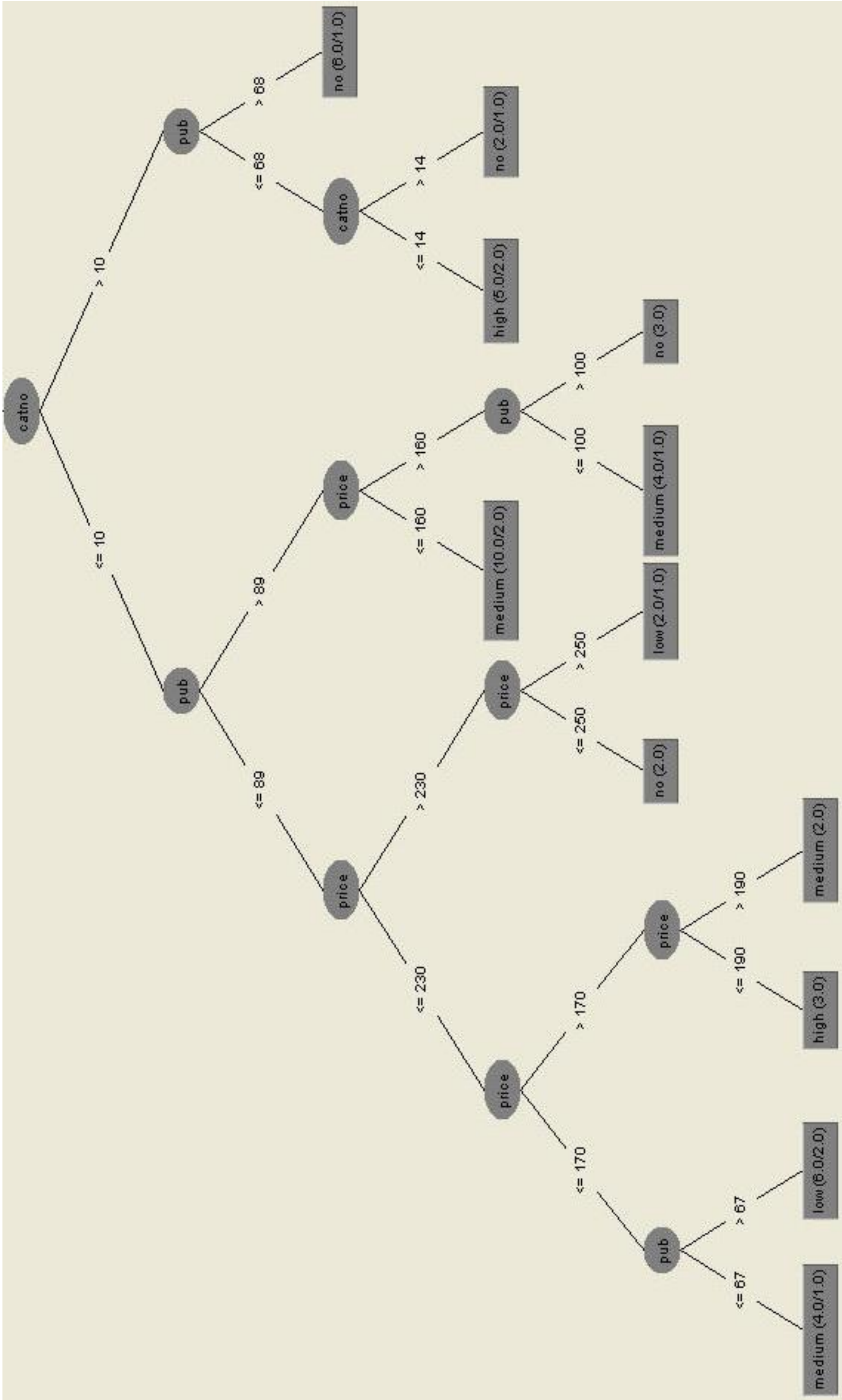


Figure C.3 Shown subtree that represent node 1.1 in Figure C.2.

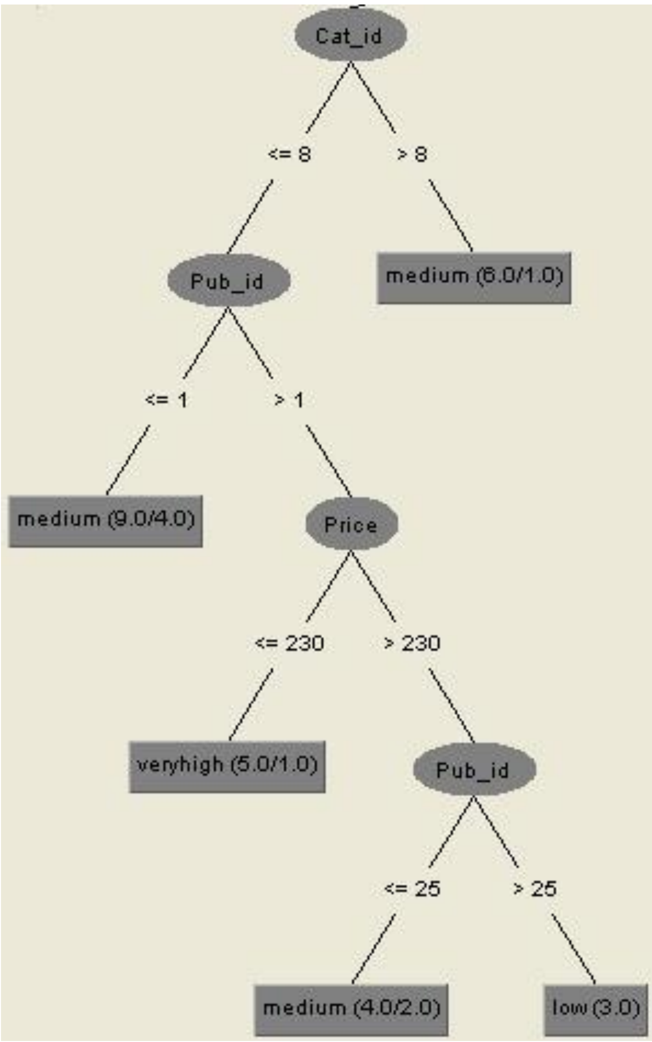


Figure C.4 Shown subtree that represent node 2 in Figure C.1.

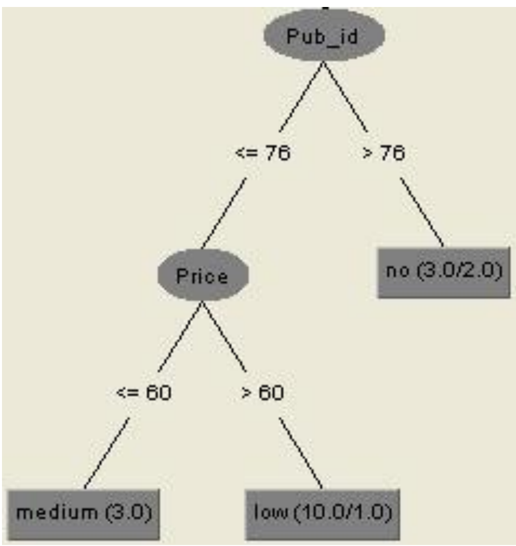


Figure C.5 Shown subtree that represent node 3 in Figure C.1.

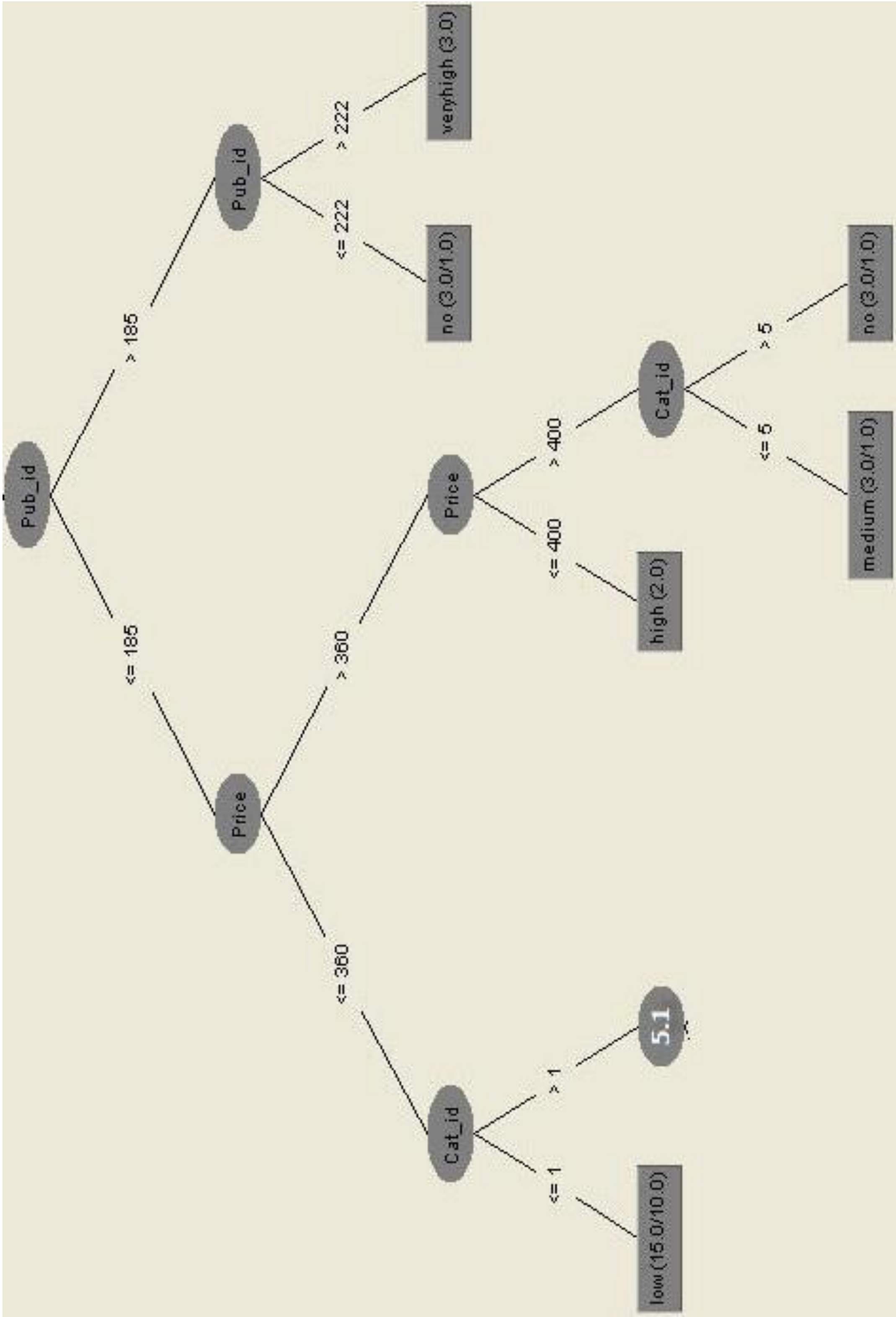


Figure C.6 Shown subtree that represent node 5 in Figure C.1.

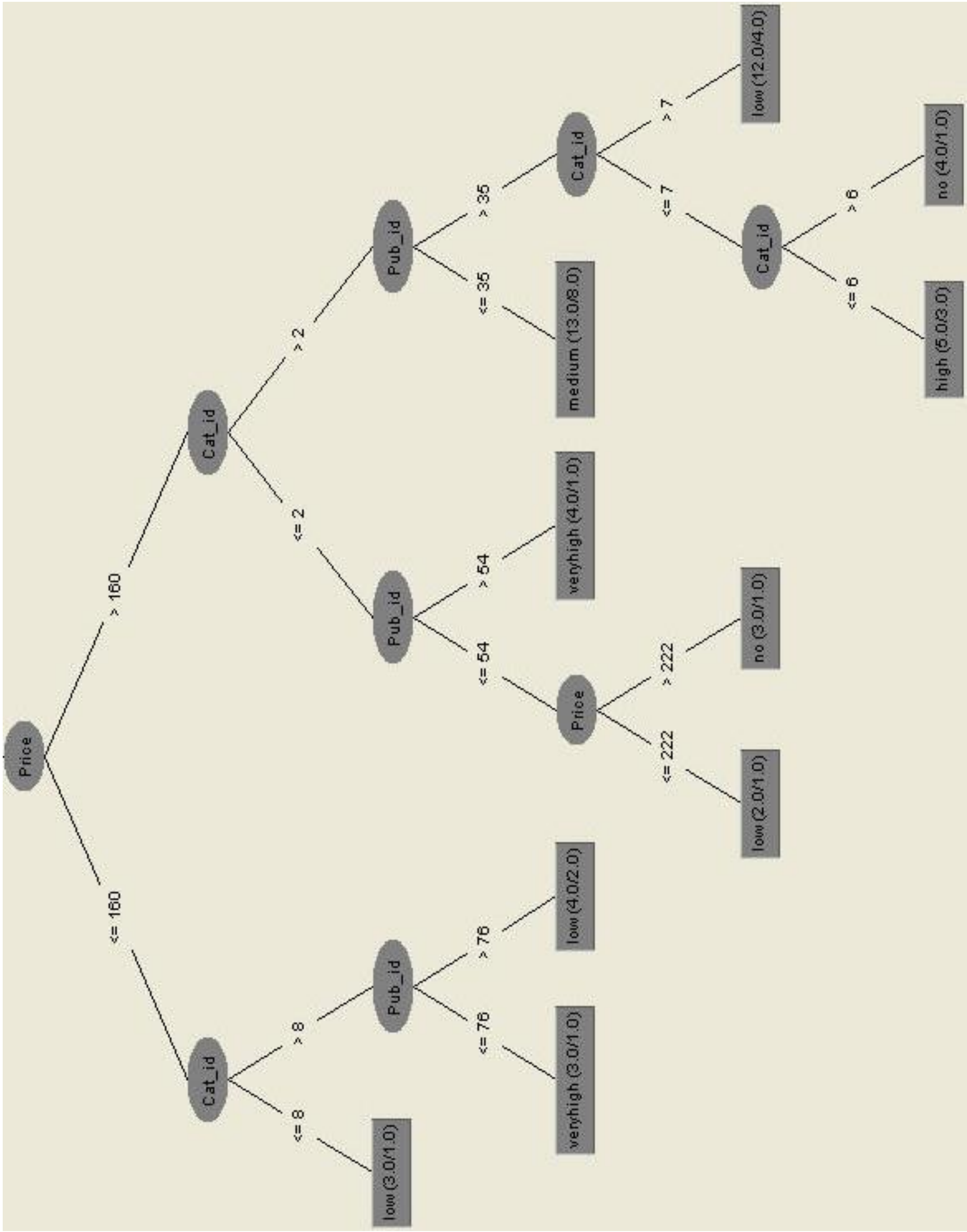


Figure C.7 Shown subtree that represent node 5.1 in Figure C.6.

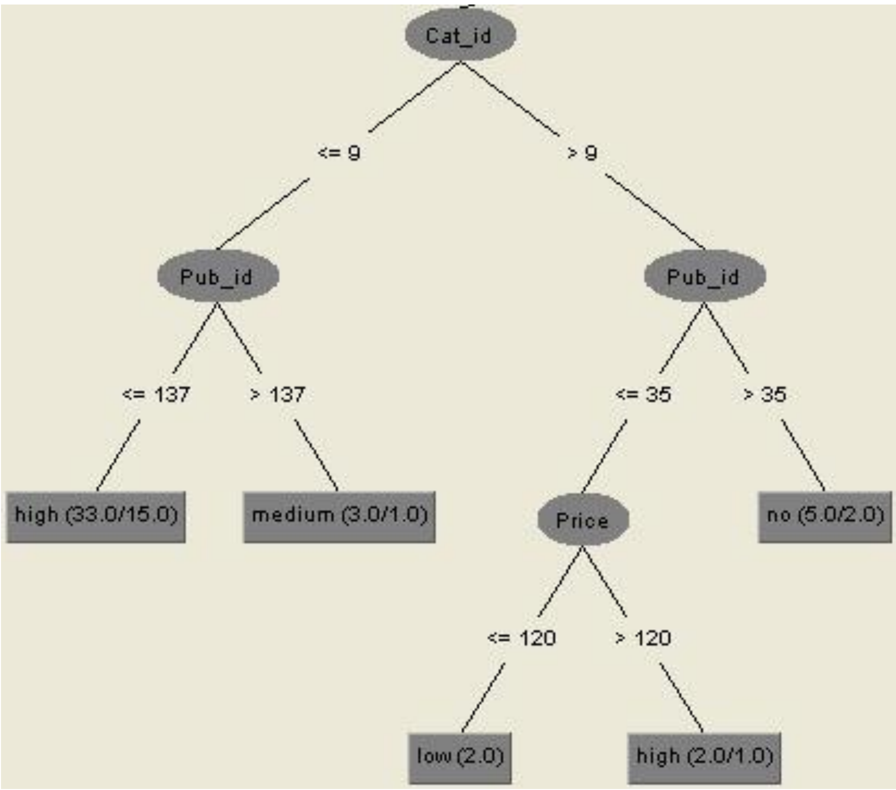


Figure C.8 Shown subtree that represent node 4 in Figure C.1.

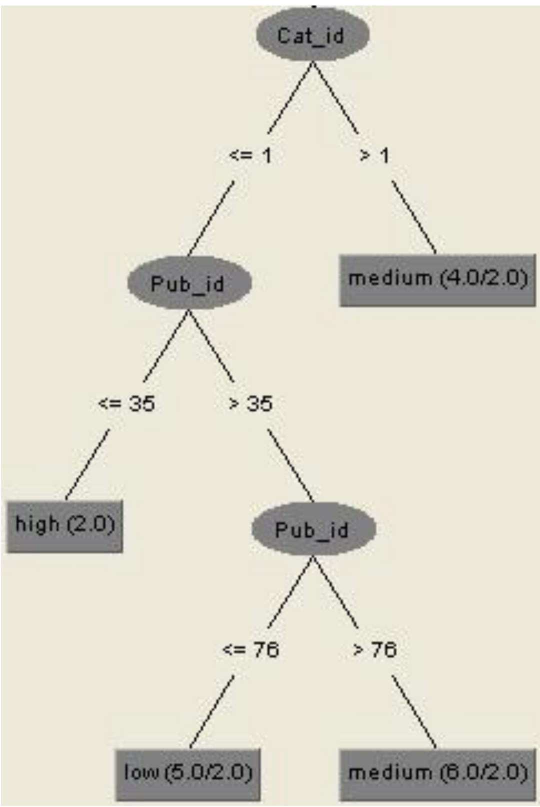


Figure C.9 Shown subtree that represent node 6 in Figure C.1.

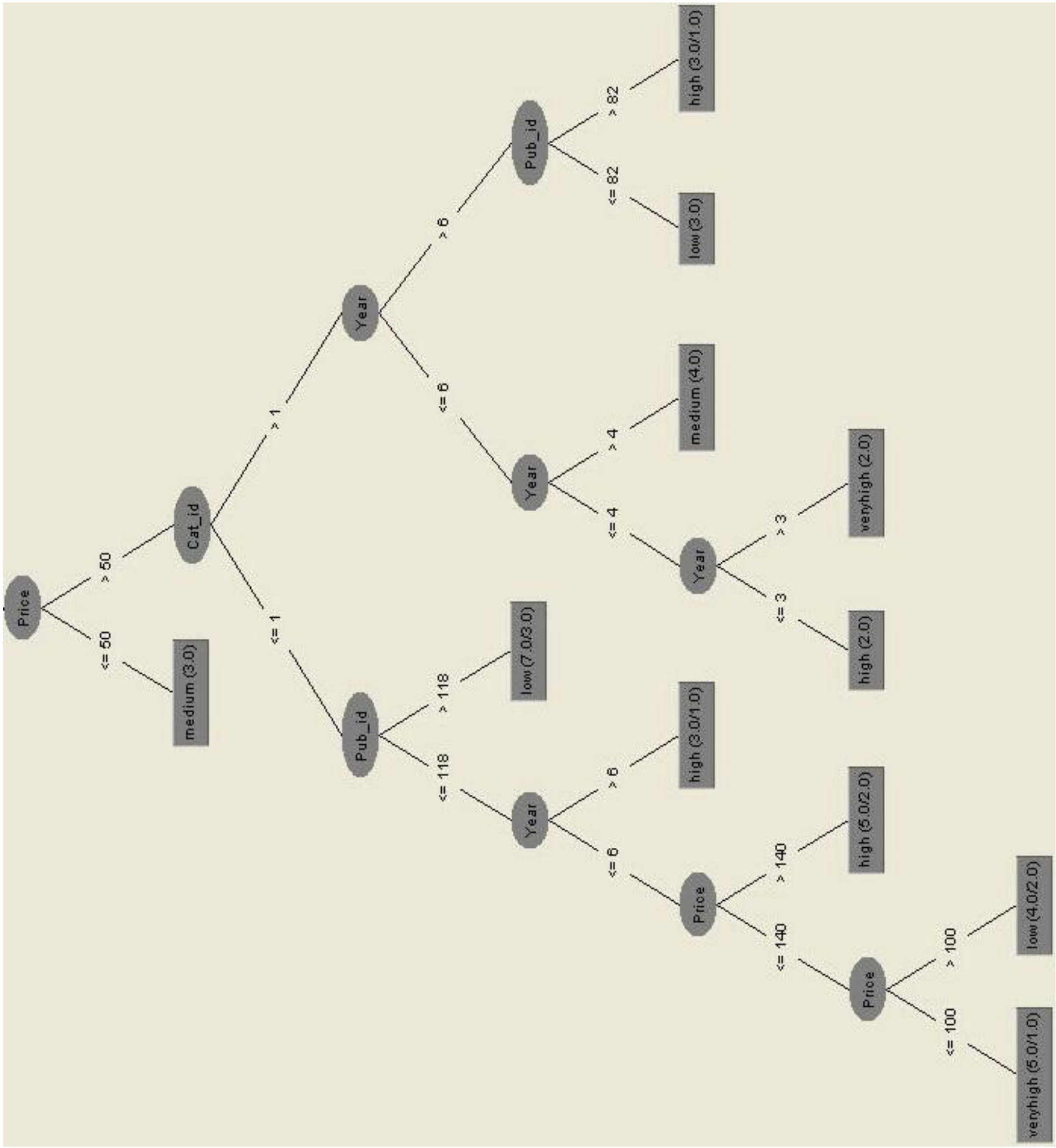


Figure C.10 Shown subtree that represent node 7 in Figure C.1.

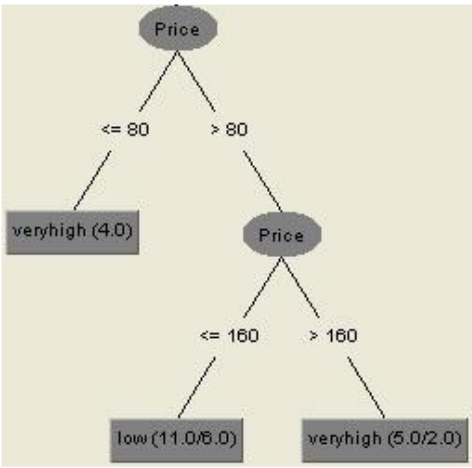


Figure C.11 Shown subtree that represent node 8 in Figure C.1.

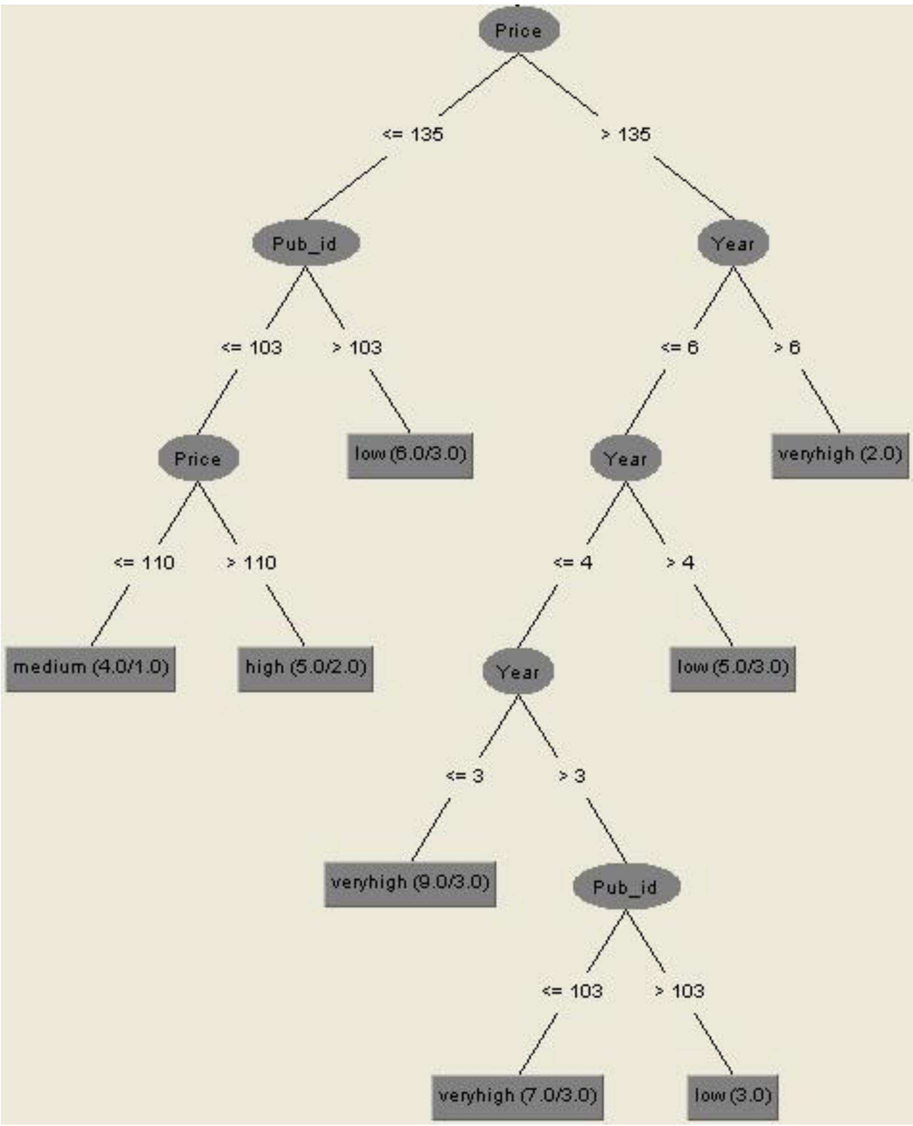


Figure C.12 Shown subtree that represent node 9 in Figure C.1.

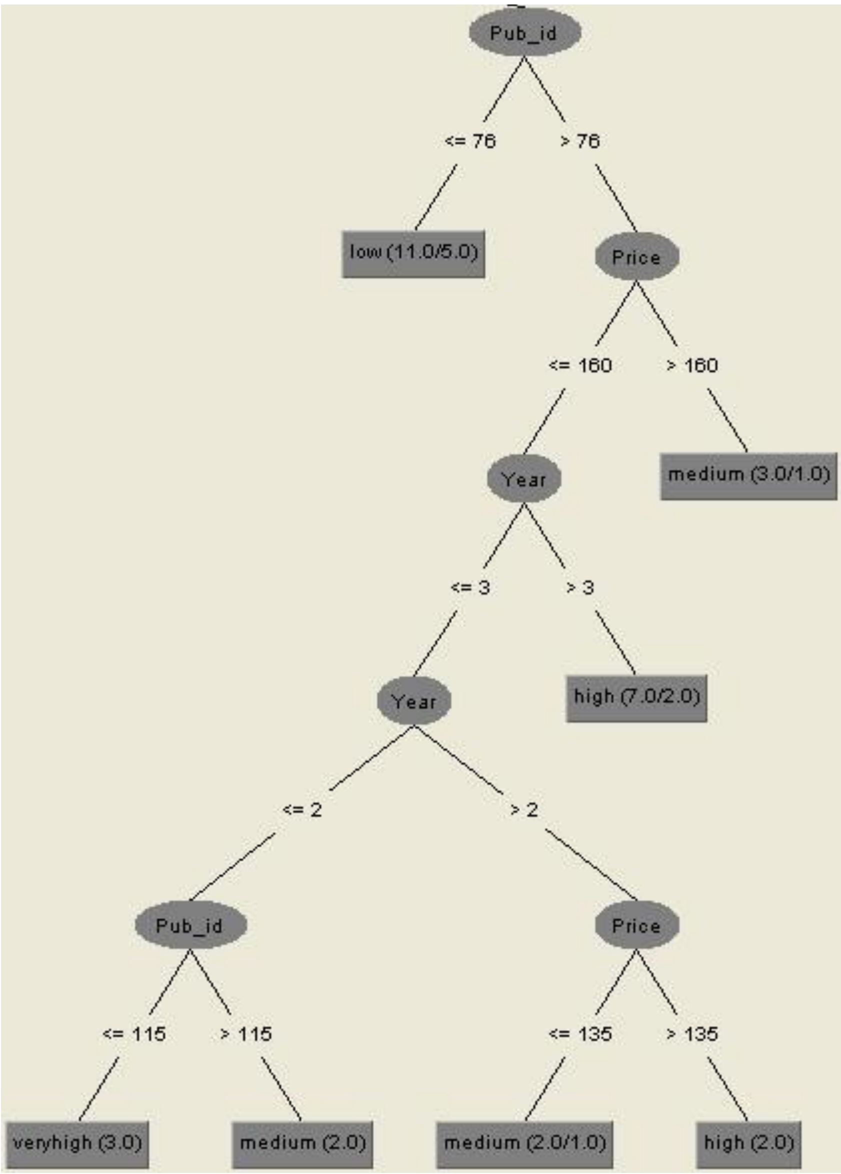


Figure C.13 Shown subtree that represent node 10 in Figure C.1.

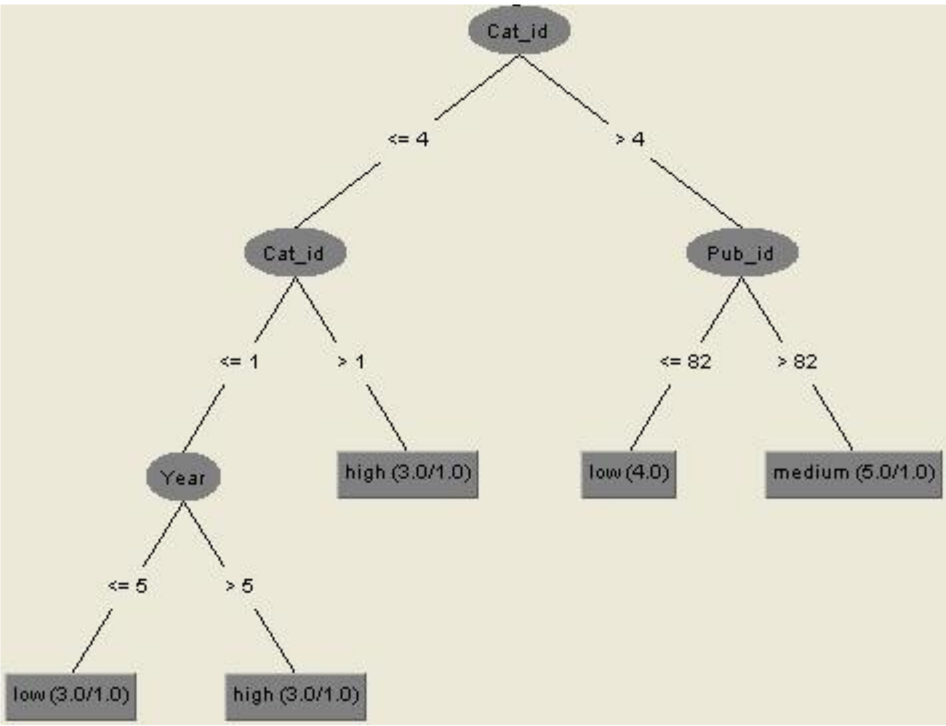


Figure C.14 Shown subtree that represent node 11 in Figure C.1.

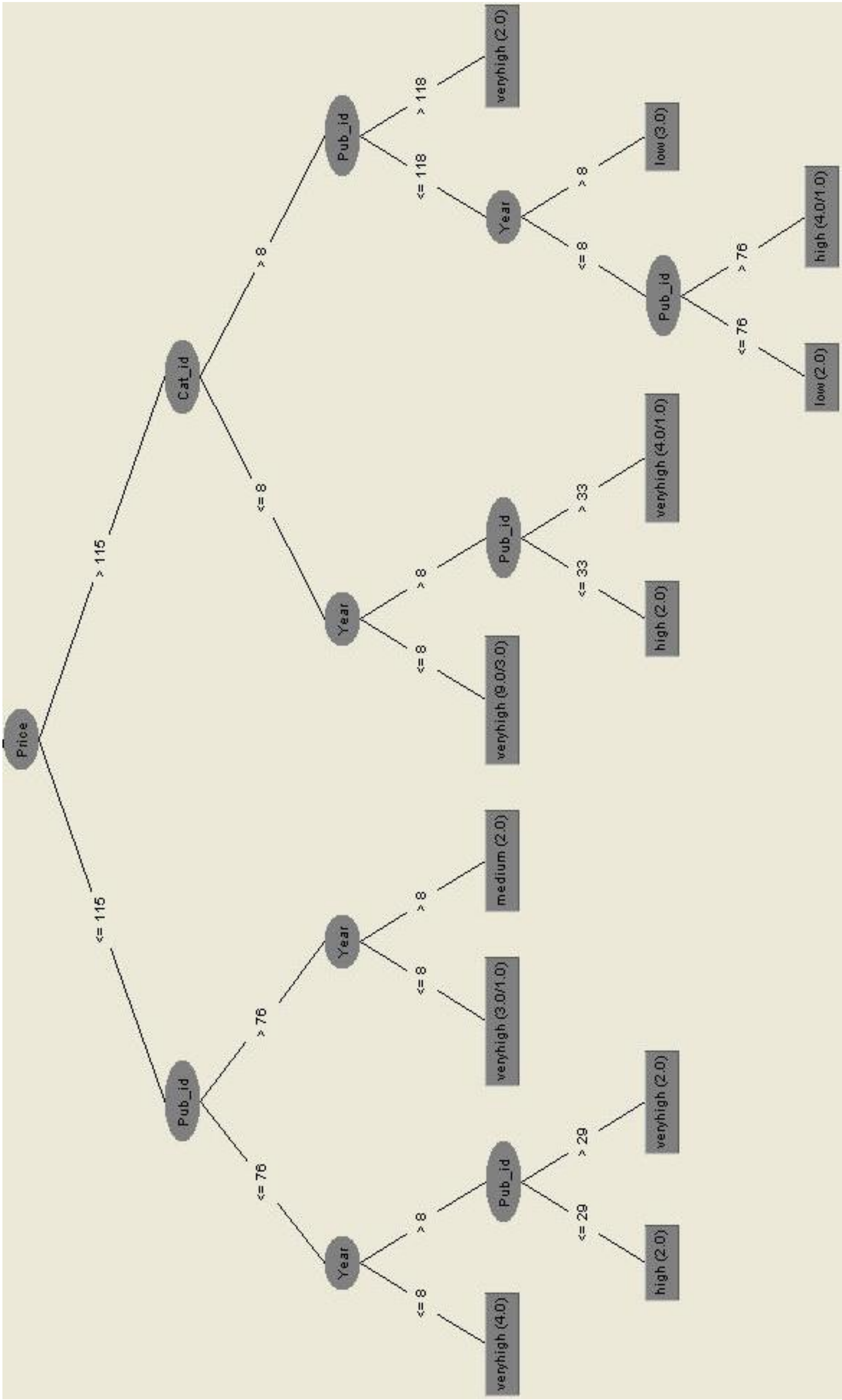


Figure C.15 Shown subtree that represent node 12 in Figure C.1.

1.2 Decision Tree of Store ID 30121

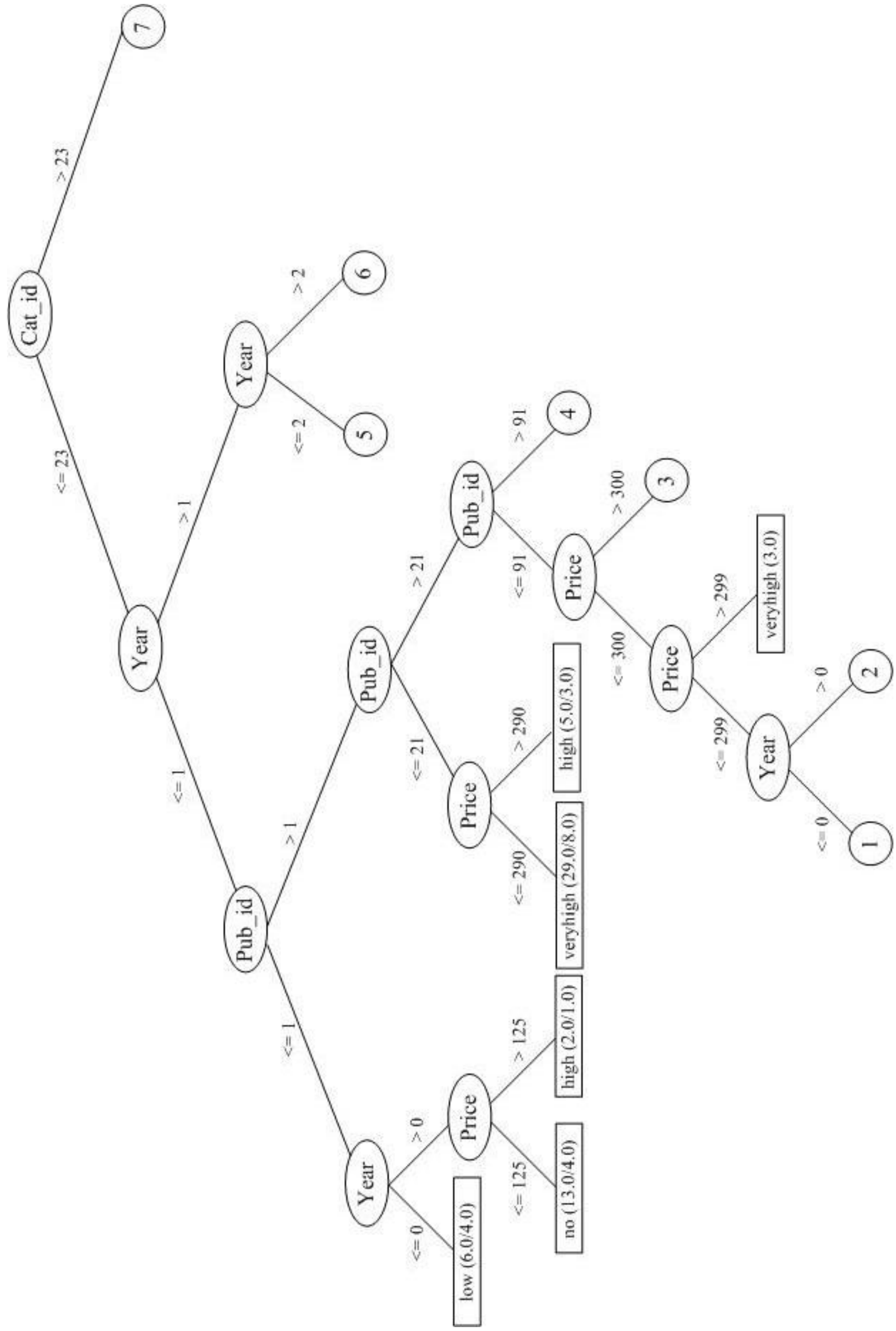


Figure C.16 Decision Tree of Store ID 30121.

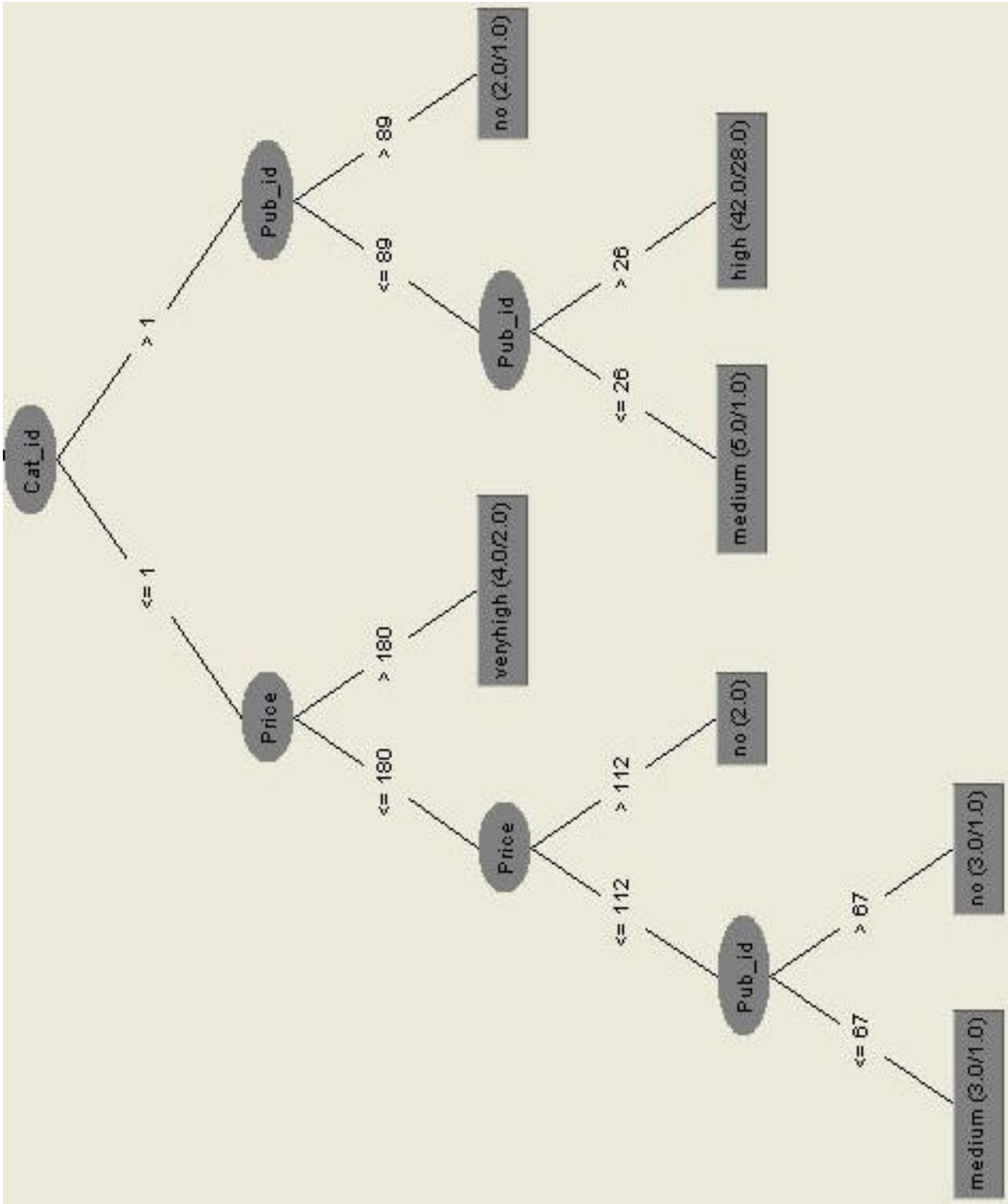


Figure C.17 Shown subtree that represent node 1 in Figure C.16.

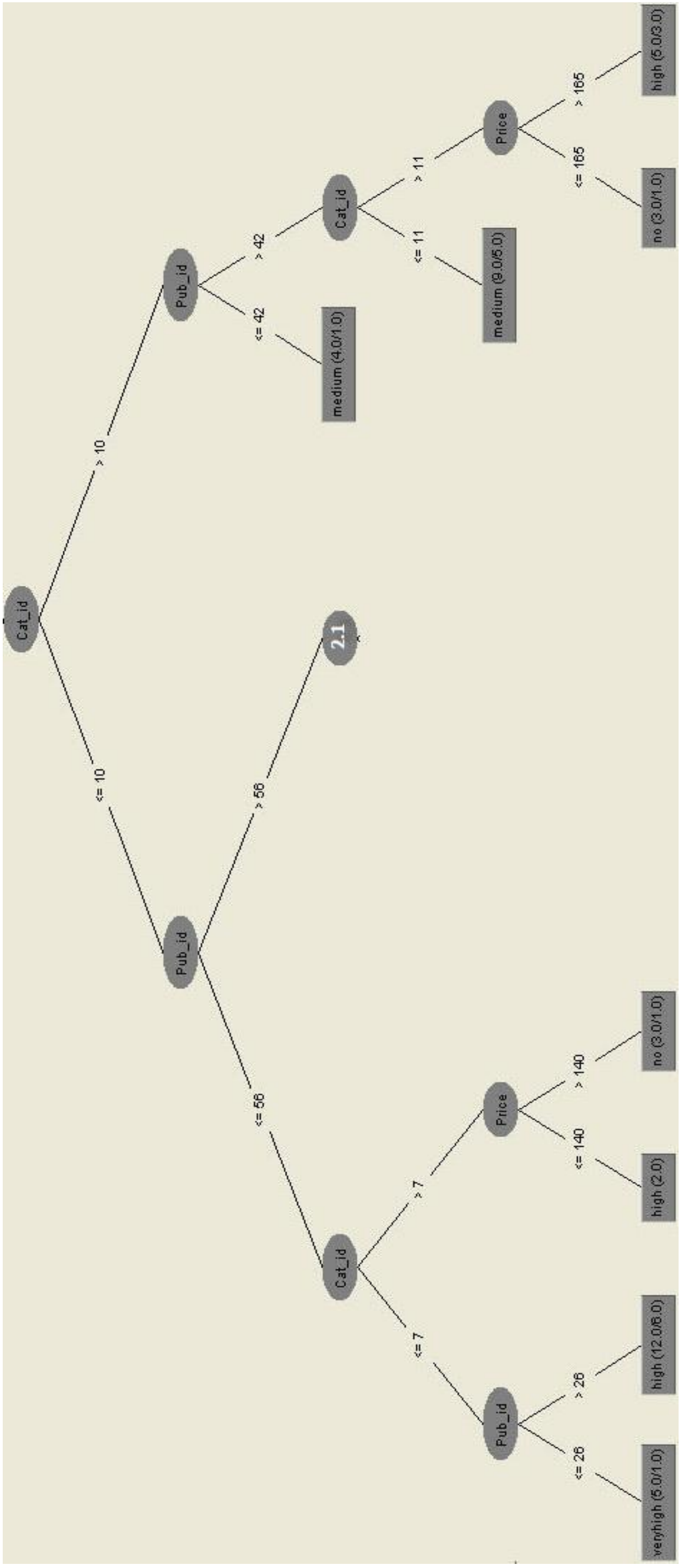


Figure C.18 Shown subtree that represent node 2 in Figure C.16.

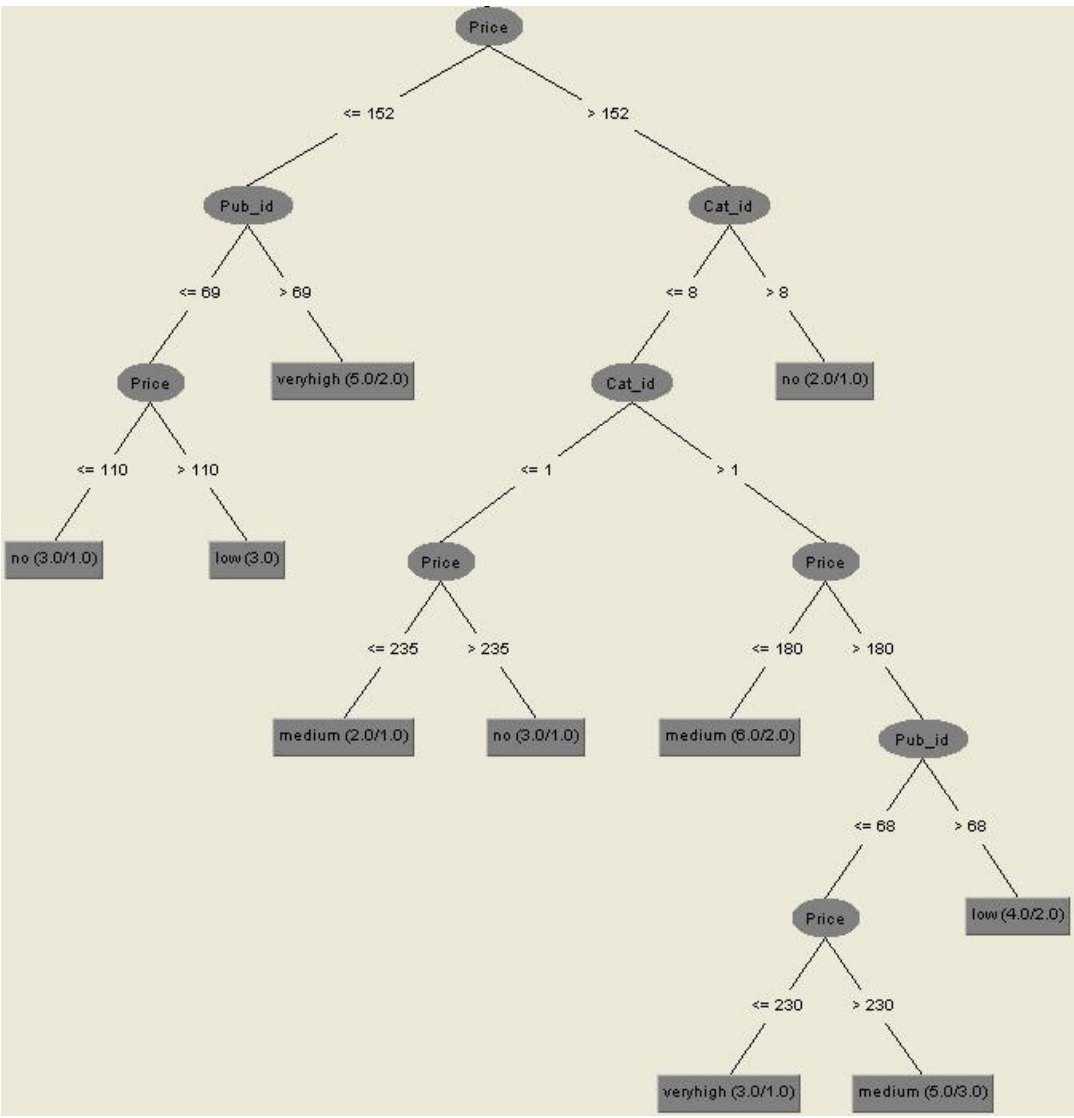


Figure C.19 Shown subtree that represent node 2.1 in Figure C.18.

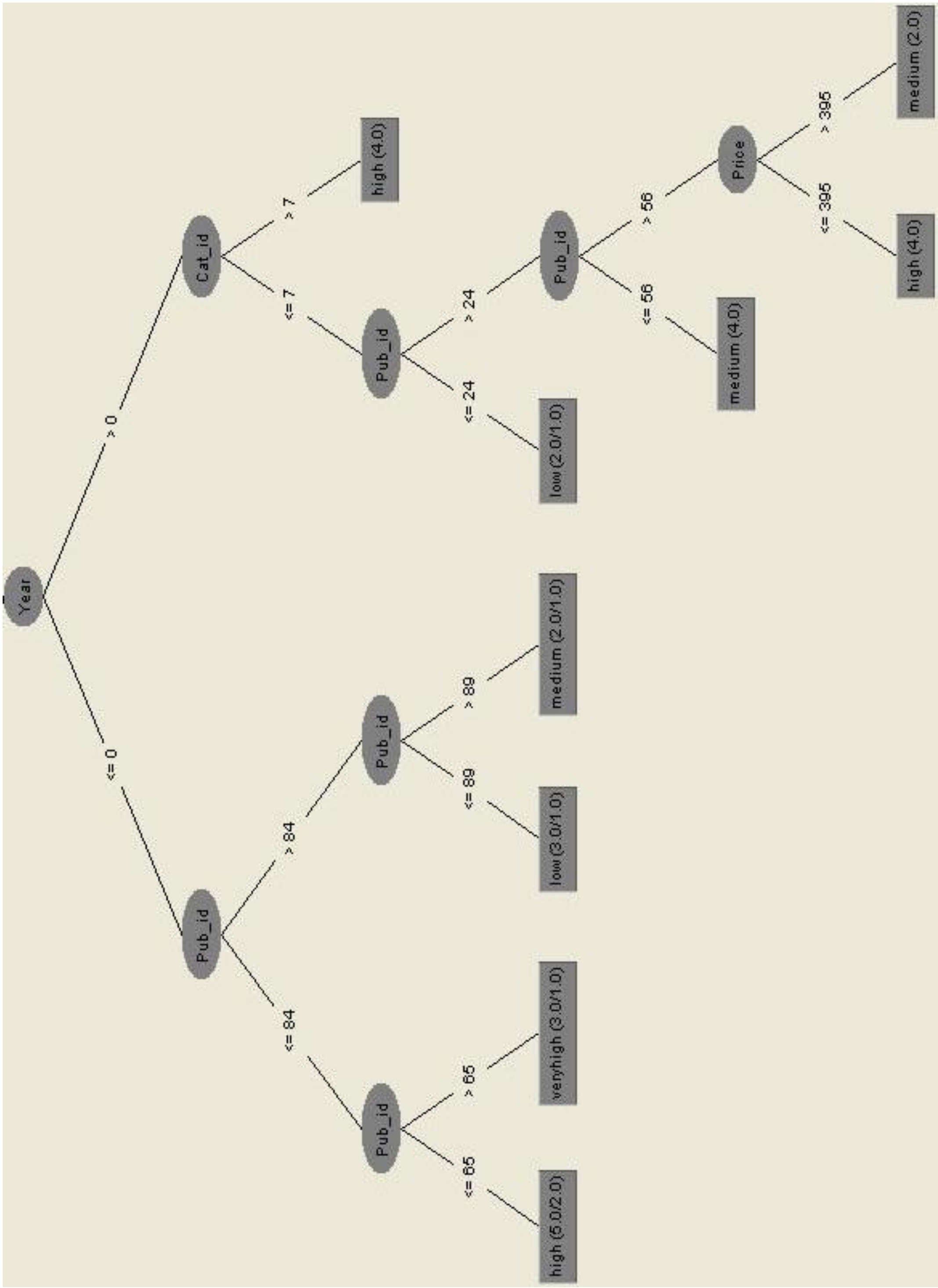


Figure C.20 Shown subtree that represent node 3 in Figure C.16.

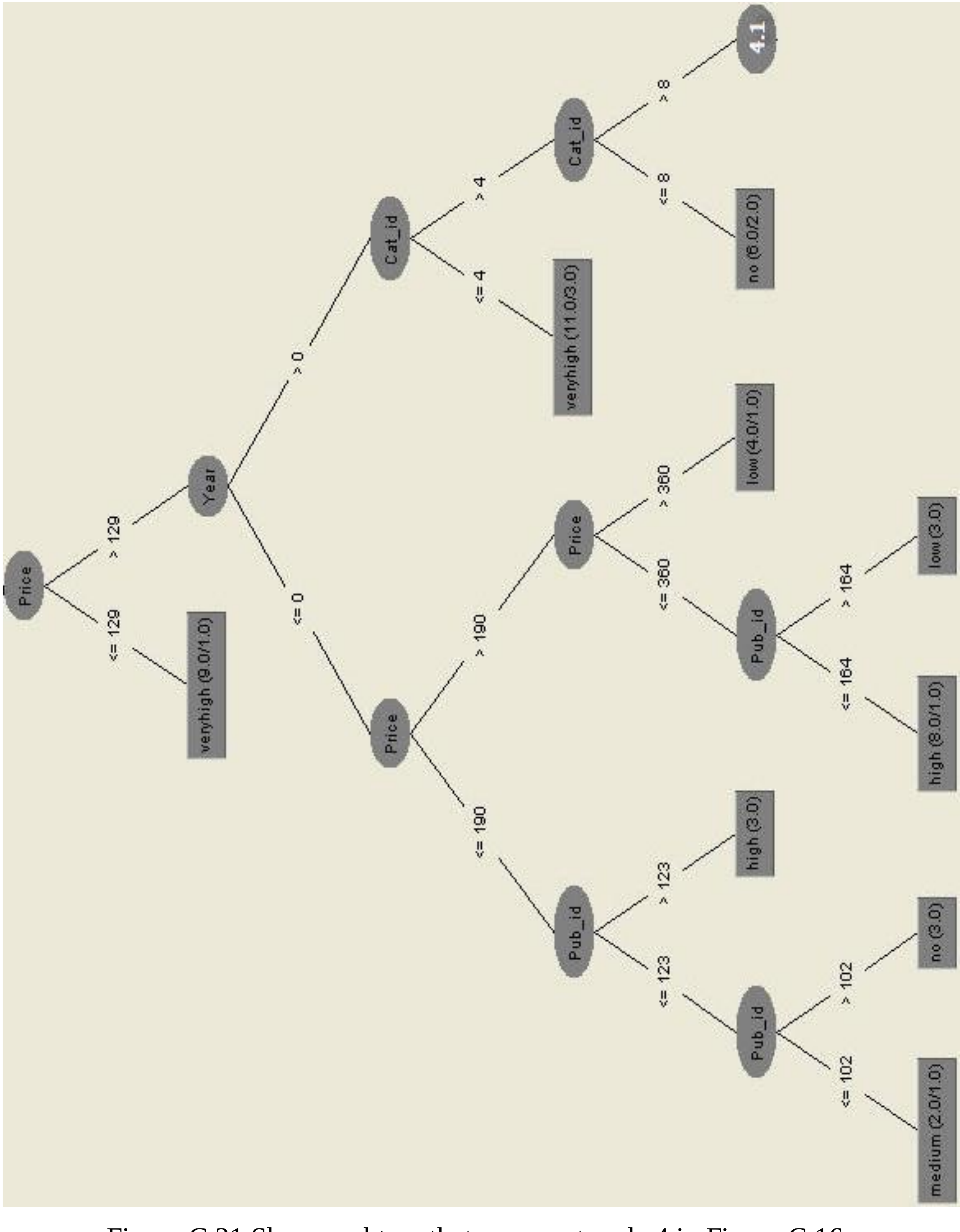


Figure C.21 Shown subtree that represent node 4 in Figure C.16.

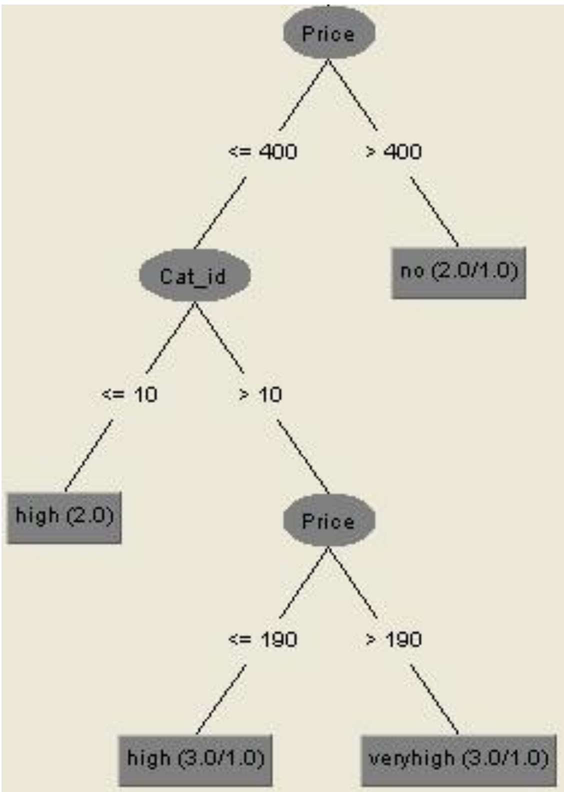


Figure C.22 Shown subtree that represent node 4.1 in Figure C.21.

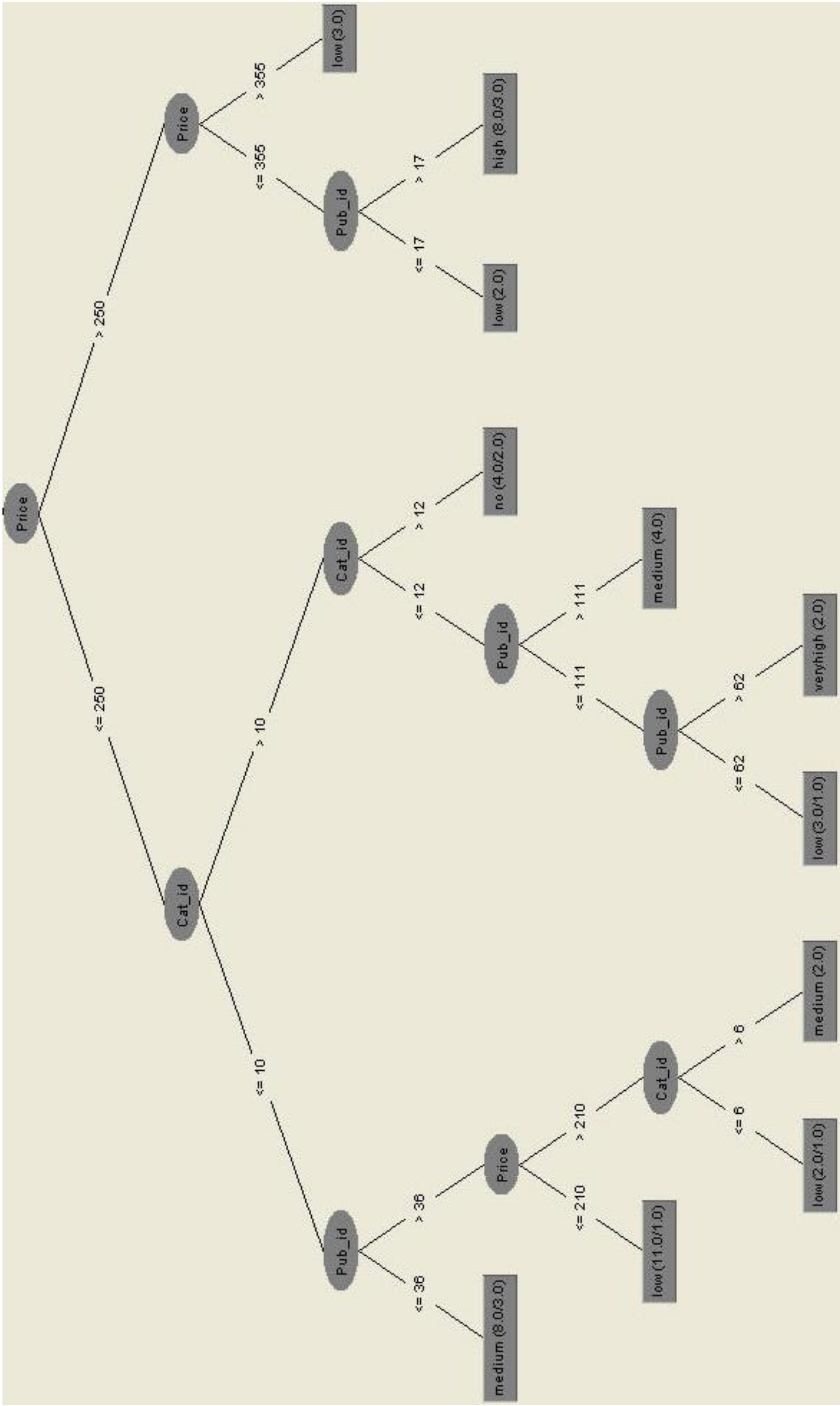


Figure C.23 Shown subtree that represent node 5 in Figure C.16.

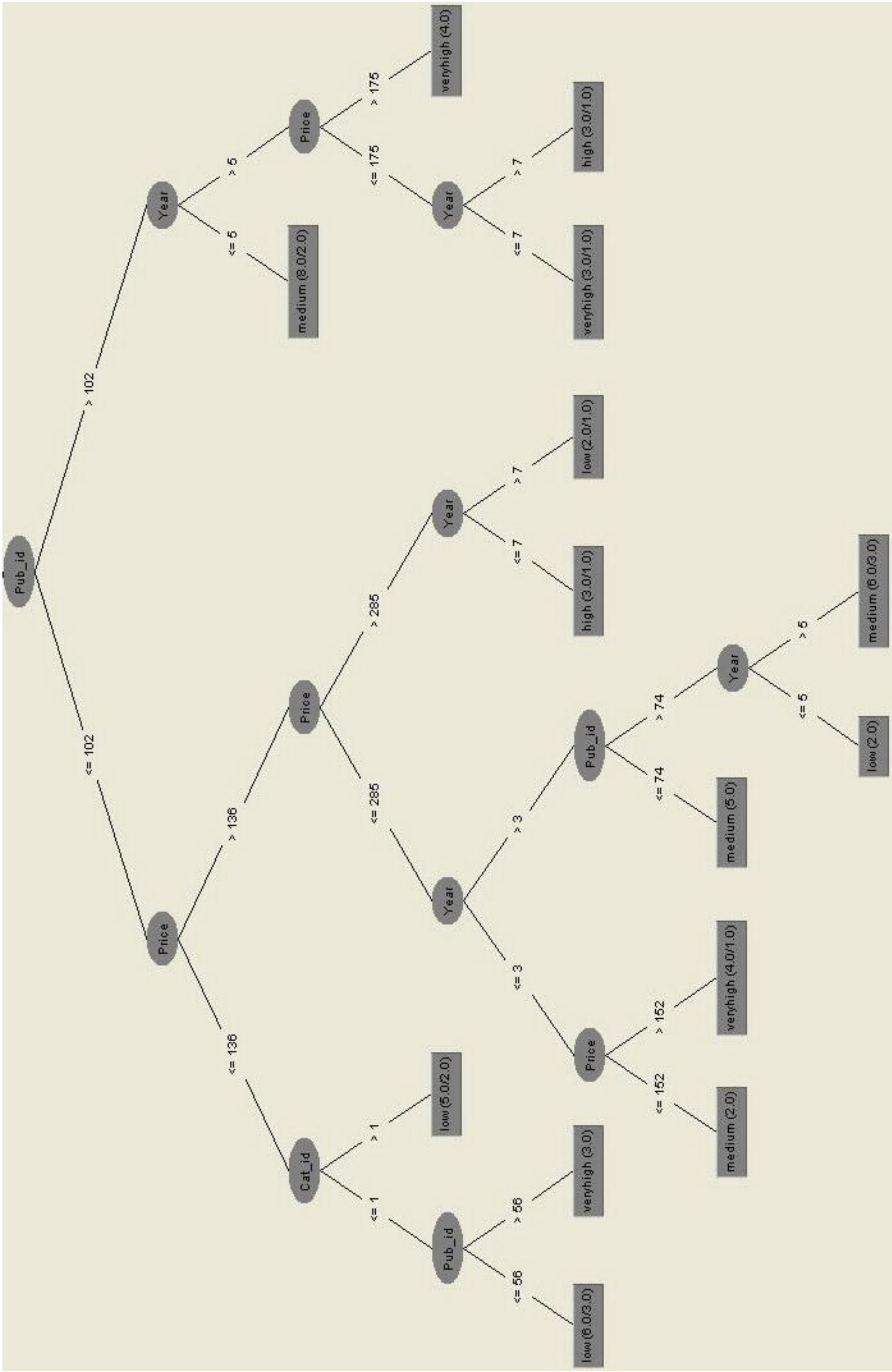


Figure C.24 Shown subtree that represent node 6 in Figure C.16.

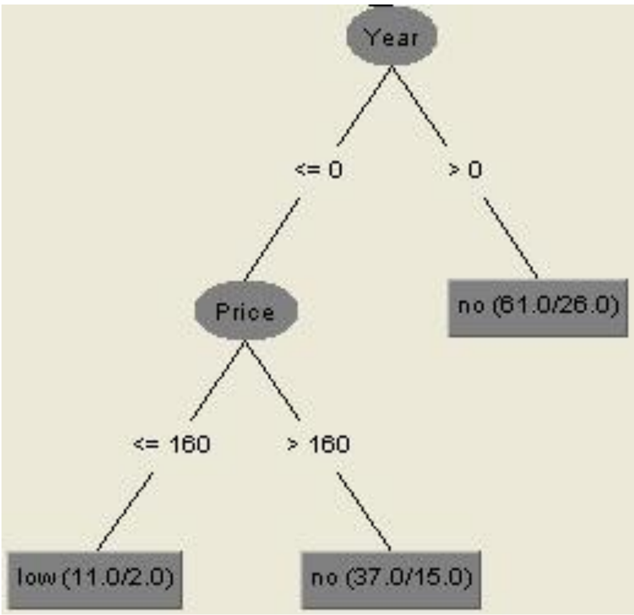


Figure C.25 Shown subtree that represent node 7 in Figure C.16.

1.3 Decision Tree of Store ID 83205

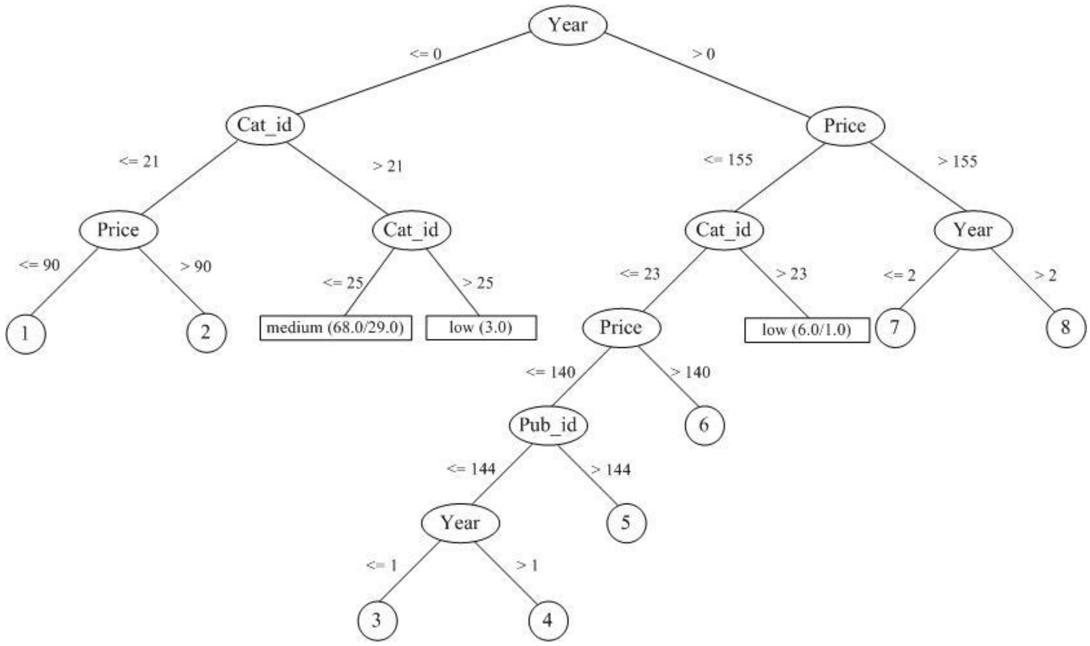


Figure C.26 Decision Tree of Store ID 83205.

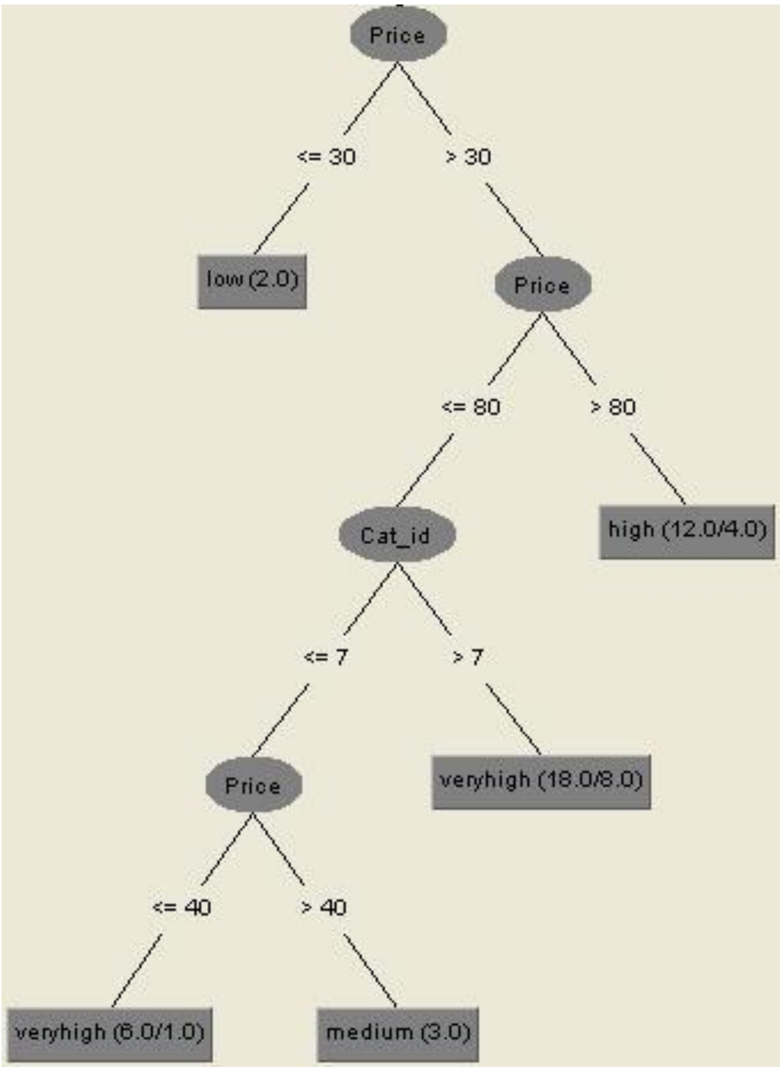


Figure C.27 Shown subtree that represent node 1 in Figure C.26.

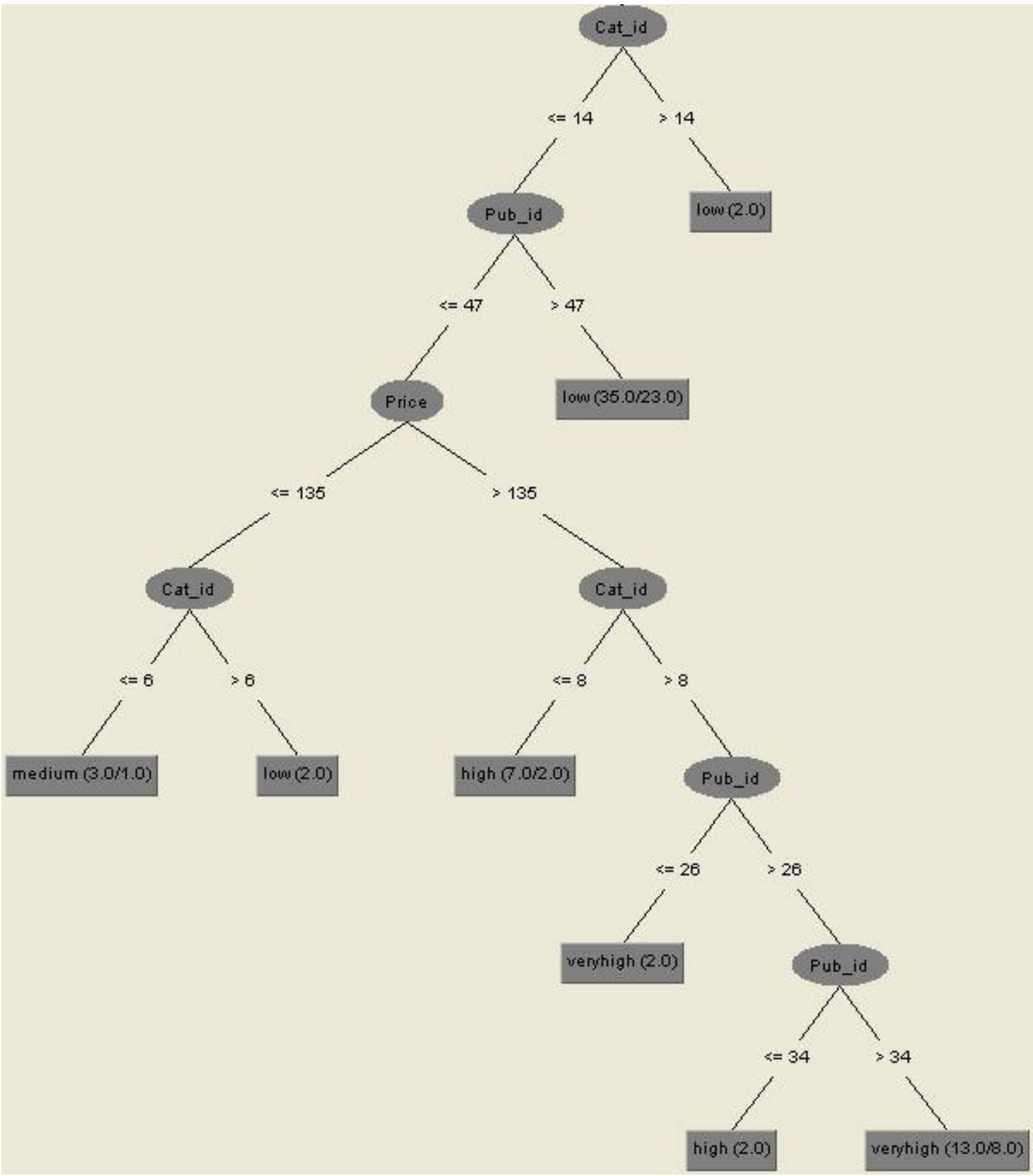


Figure C.28 Shown subtree that represent node 2 in Figure C.26.

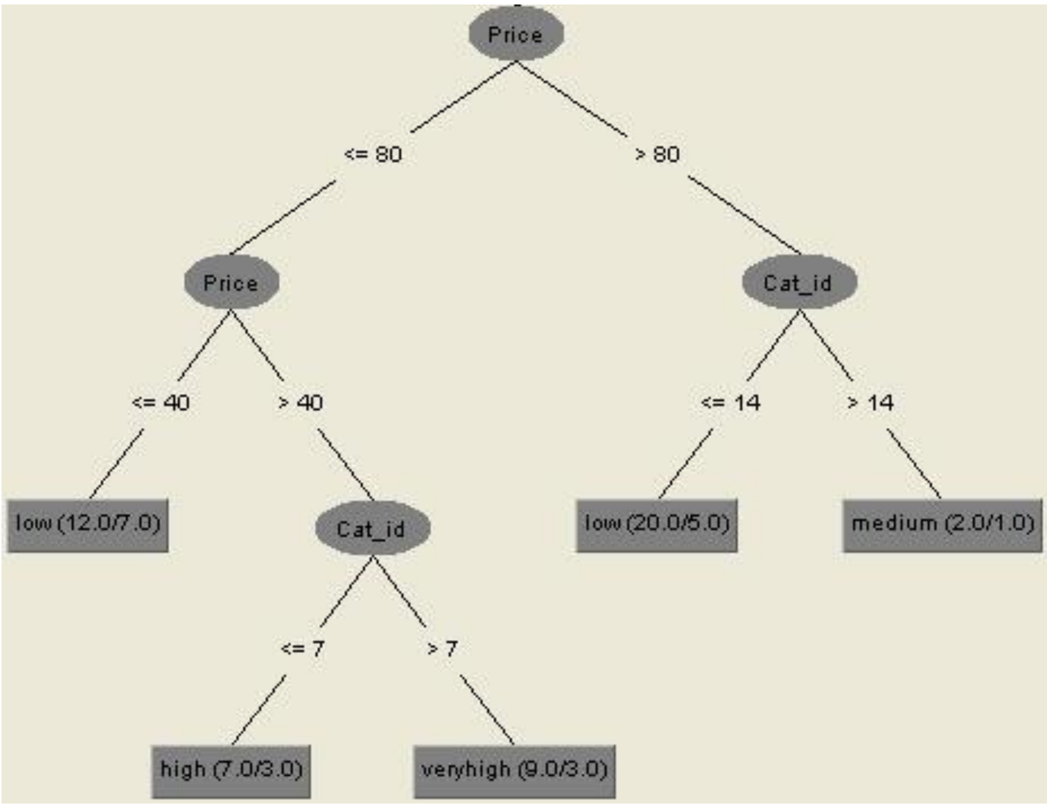


Figure C.29 Shown subtree that represent node 3 in Figure C.26.

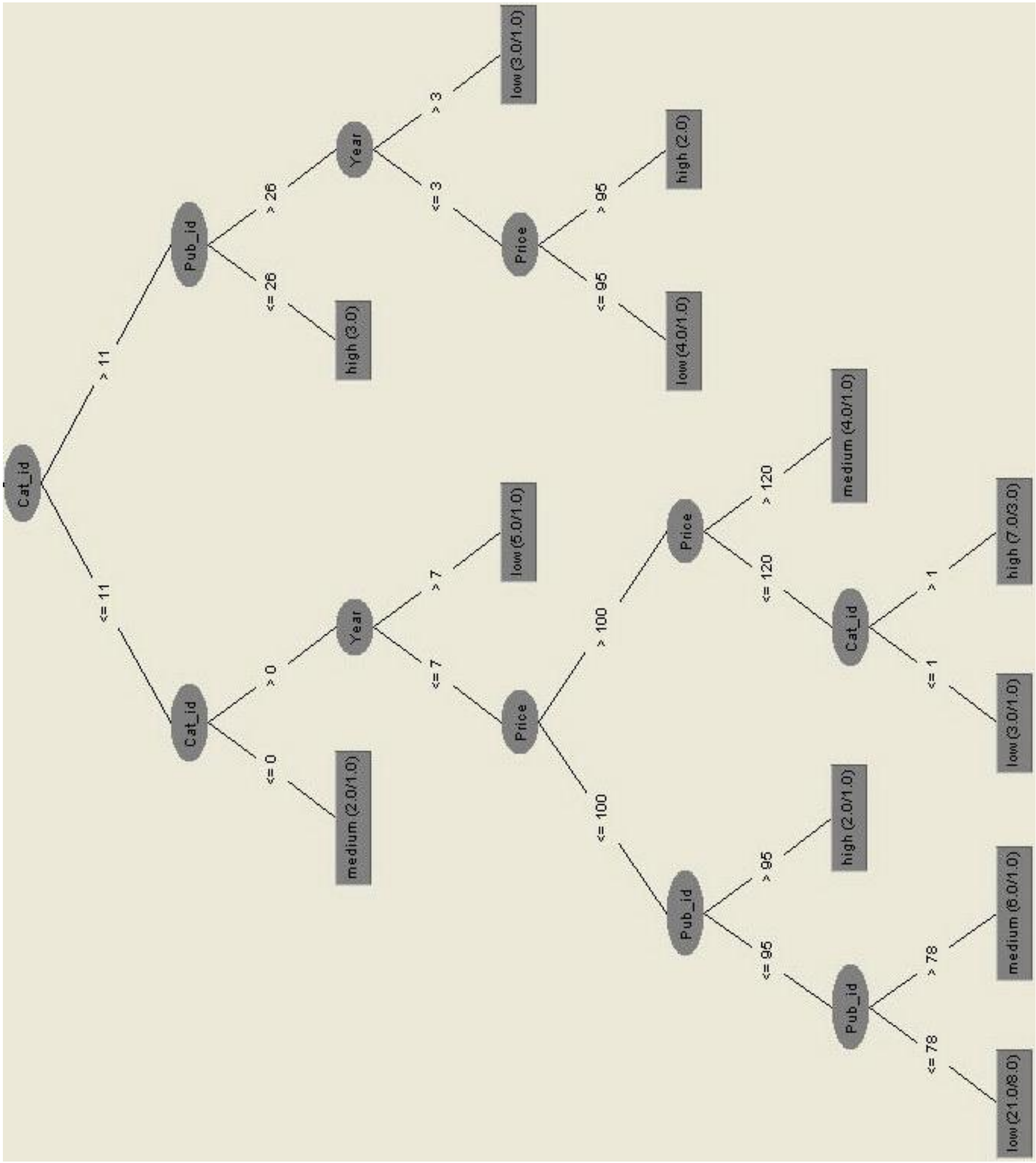


Figure C.30 Shown subtree that represent node 4 in Figure C.26.

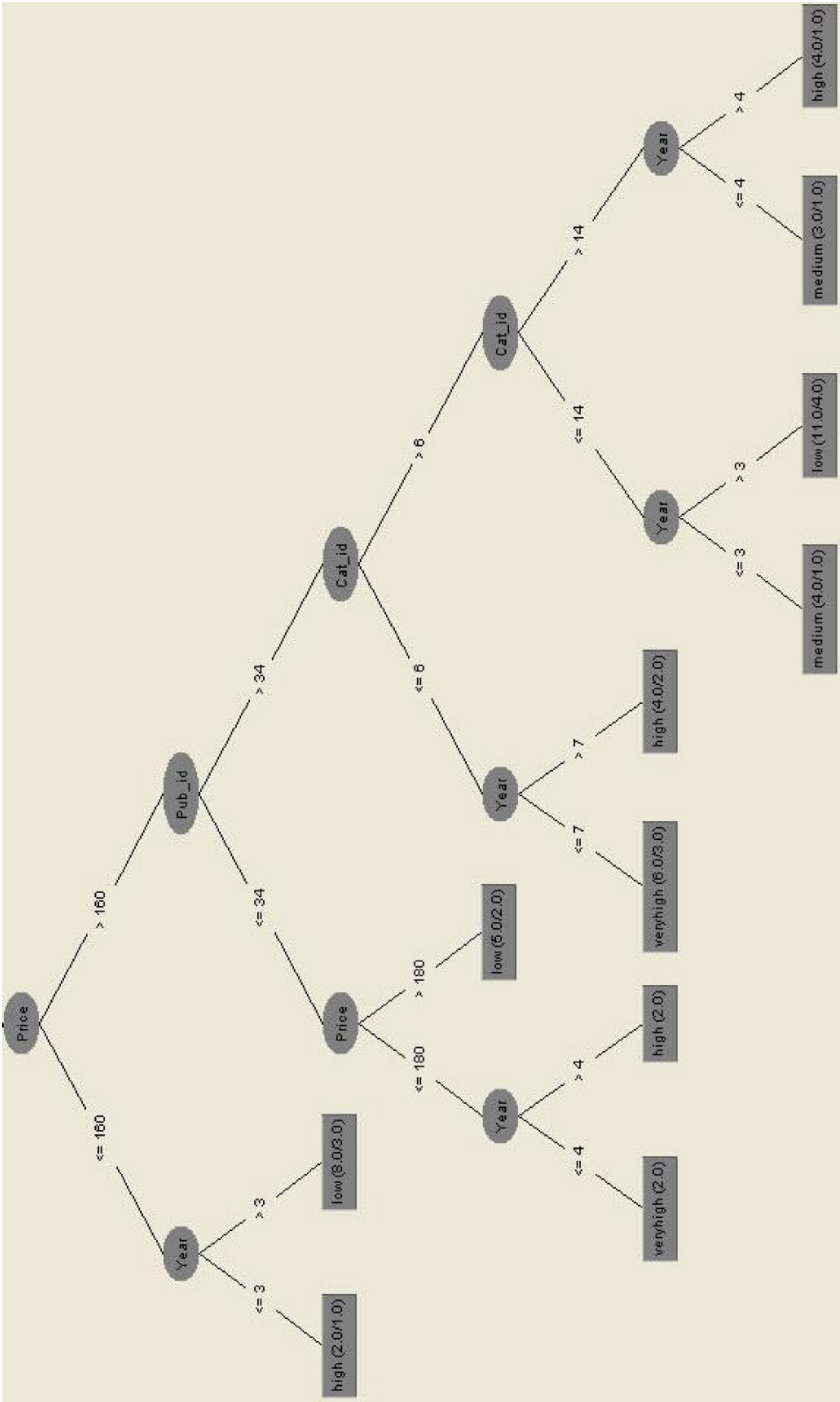


Figure C.31 Shown subtree that represent node 5 in Figure C.26.

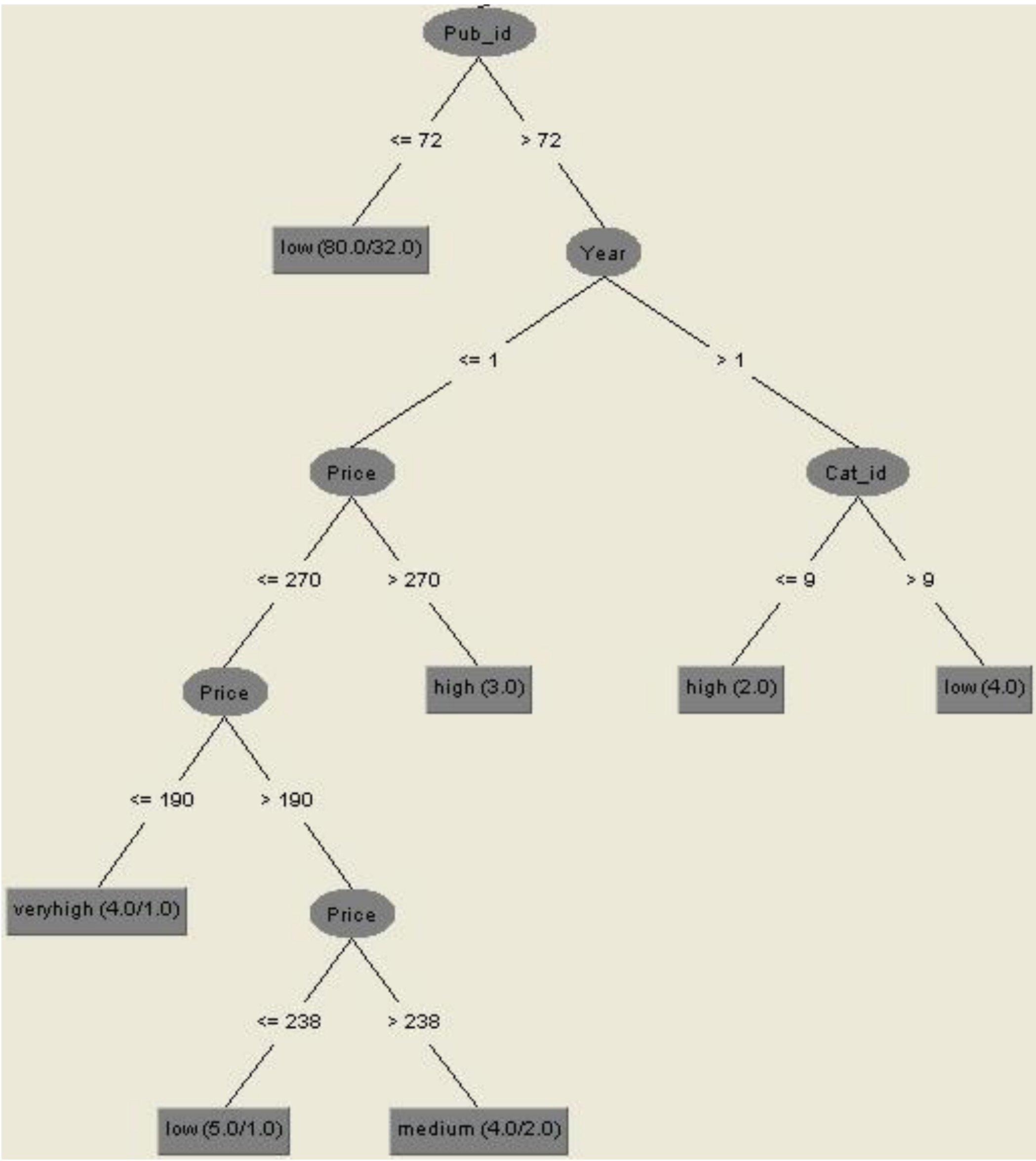


Figure C.32 Shown subtree that represent node 6 in Figure C.26.

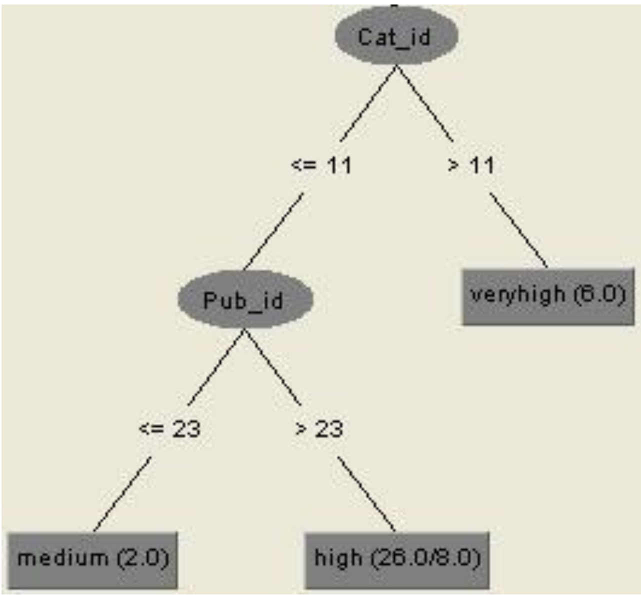


Figure C.33 Shown subtree that represent node 7 in Figure C.26.

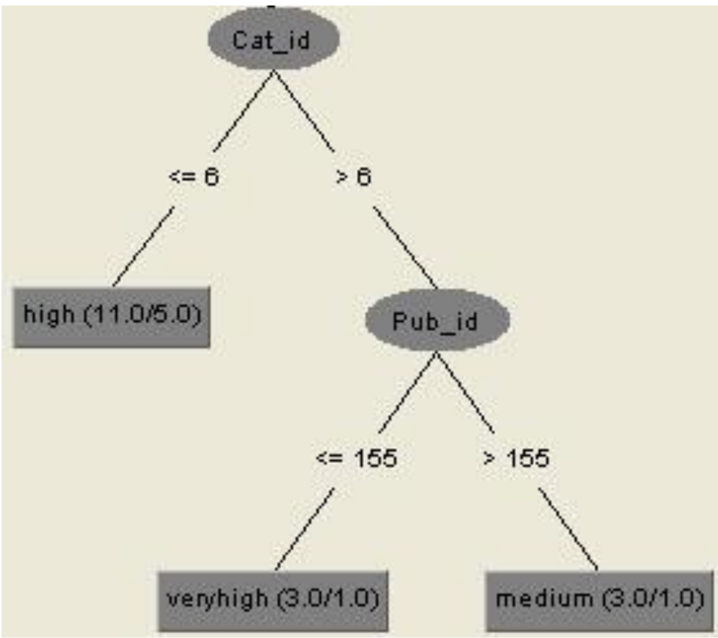


Figure C.34 Shown subtree that represent node 8 in Figure C.26.

2. Data Type is Numeric Data

2.1. Decision Tree of Store ID 11218

Table C.1 Decision tree of store id 11218 in conditions form.

Root Node (L1)	L2	L3	L4	L5	L6	L7
Pub_id = 1						
	Year <= 0: medium (9.0/5.0)					
	Year > 0					
		Year <= 1: veryhigh (10.0/8.0)				
		Year > 1: low (10.0/4.0)				
Pub_id = 8: veryhigh (6.0/6.0)						
Pub_id = 11: no (2.0/1.0)						
Pub_id = 13						
	Year <= 7: low (4.0/3.0)					
	Year > 7					
		Year <= 8: veryhigh (4.0/4.0)				
		Year > 8: high (4.0/4.0)				
Pub_id = 19: high (2.0/1.0)						
Pub_id = 21: medium (15.0/6.0)						
Pub_id = 22						
	Year <= 2					
		Cat_id = 11: veryhigh (7.0/3.0)				
		Cat_id = 112: high (2.0/1.0)				
	Year > 2: low (3.0/2.0)					
Pub_id = 23: no (1.0/1.0)						
Pub_id = 24						
	Year <= 1: medium (4.0/3.0)					
	Year > 1: low (2.0/2.0)					
Pub_id = 25						
	Year <= 7					
		Year <= 1: medium (2.0/1.0)				
		Year > 1: high (4.0/3.0)				
	Year > 7: low (2.0/2.0)					
Pub_id = 26						
	Year <= 0: low (2.0/1.0)					
	Year > 0					
		Year <= 1: medium (2.0/2.0)				
		Year > 1: veryhigh (4.0/2.0)				
Pub_id = 27: low (3.0/1.0)						
Pub_id = 29: low (2.0/1.0)						
Pub_id = 31: no (2.0/1.0)						
Pub_id = 33						
	Year <= 0: low (2.0/2.0)					
	Year > 0: veryhigh (4.0/3.0)					
Pub_id = 35						
	Year <= 0: no (2.0/1.0)					
	Year > 0: low (2.0/2.0)					
Pub_id = 41						
	Year <= 1: high (2.0/2.0)					
	Year > 1: no (5.0/2.0)					

Table C.1 Decision tree of store id 11218 in conditions form. (Cont.)

<table>
<tr><th>Root Node (L1)</th><th>L2</th><th>L3</th><th>L4</th><th>L5</th><th>L6</th><th>L7</th></tr>
<tr><td colspan="7">Pub_id = 47</td></tr>
<tr><td></td><td colspan="6">| Year <= 7</td></tr>
<tr><td></td><td></td><td colspan="5">| | Price <= 40: medium (4.0/4.0)</td></tr>
<tr><td></td><td></td><td colspan="5">| | Price > 40</td></tr>
<tr><td></td><td></td><td></td><td colspan="4">| | | Year <= 0: medium (4.0/3.0)</td></tr>
<tr><td></td><td></td><td></td><td colspan="4">| | | Year > 0</td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="3">| | | | Year <= 6</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td colspan="2">| | | | | Price <= 140</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>| | | | | | Price <= 110: veryhigh (7.0/3.0)</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>| | | | | | Price > 110: low (2.0/2.0)</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td colspan="2">| | | | | Price > 140: high (10.0/6.0)</td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="3">| | | | Year > 6: high (4.0/2.0)</td></tr>
<tr><td></td><td colspan="6">| Year > 7: veryhigh (2.0/2.0)</td></tr>
<tr><td colspan="7">Pub_id = 48</td></tr>
<tr><td></td><td colspan="6">| Year <= 2: no (6.0/5.0)</td></tr>
<tr><td></td><td colspan="6">| Year > 2: veryhigh (6.0/3.0)</td></tr>
<tr><td colspan="7">Pub_id = 49</td></tr>
<tr><td></td><td colspan="6">| Year <= 0: low (9.0/8.0)</td></tr>
<tr><td></td><td colspan="6">| Year > 0</td></tr>
<tr><td></td><td></td><td colspan="5">| | Year <= 1: high (11.0/7.0)</td></tr>
<tr><td></td><td></td><td colspan="5">| | Year > 1: veryhigh (11.0/9.0)</td></tr>
<tr><td colspan="7">Pub_id = 50: no (2.0/2.0)</td></tr>
<tr><td colspan="7">Pub_id = 51: high (3.0/2.0)</td></tr>
<tr><td colspan="7">Pub_id = 53: no (1.0/1.0)</td></tr>
<tr><td colspan="7">Pub_id = 54</td></tr>
<tr><td></td><td colspan="6">| Cat_id = 11: low (2.0/1.0)</td></tr>
<tr><td colspan="7">Pub_id = 61: low (3.0/1.0)</td></tr>
<tr><td colspan="7">Pub_id = 63: low (2.0/2.0)</td></tr>
<tr><td colspan="7">Pub_id = 64: low (2.0/1.0)</td></tr>
<tr><td colspan="7">Pub_id = 65: high (4.0/2.0)</td></tr>
<tr><td colspan="7">Pub_id = 66</td></tr>
<tr><td></td><td colspan="6">| Cat_id = 2: medium (7.0/4.0)</td></tr>
<tr><td></td><td colspan="6">| Cat_id = 6</td></tr>
<tr><td></td><td></td><td colspan="5">| | Price <= 280: high (7.0/4.0)</td></tr>
<tr><td></td><td></td><td colspan="5">| | Price > 280: no (2.0/1.0)</td></tr>
<tr><td></td><td colspan="6">| Cat_id = 8: no (1.0)</td></tr>
<tr><td></td><td colspan="6">| Cat_id = 9: low (2.0/1.0)</td></tr>
<tr><td colspan="7">Pub_id = 67: high (2.0/1.0)</td></tr>
<tr><td colspan="7">Pub_id = 68</td></tr>
<tr><td></td><td colspan="6">| Year <= 2: low (19.0/6.0)</td></tr>
<tr><td></td><td colspan="6">| Year > 2: veryhigh (3.0/3.0)</td></tr>
<tr><td colspan="7">Pub_id = 69: no (2.0/2.0)</td></tr>
<tr><td colspan="7">Pub_id = 70: medium (1.0/1.0)</td></tr>
<tr><td colspan="7">Pub_id = 71: no (2.0/2.0)</td></tr>
<tr><td colspan="7">Pub_id = 72: low (2.0/2.0)</td></tr>
<tr><td colspan="7">Pub_id = 76: no (3.0/1.0)</td></tr>
<tr><td colspan="7">Pub_id = 81: no (3.0/2.0)</td></tr>
<tr><td colspan="7">Pub_id = 82</td></tr>
<tr><td></td><td colspan="6">| Year <= 0: no (4.0/4.0)</td></tr>
<tr><td></td><td colspan="6">| Year > 0: low (4.0/2.0)</td></tr>
</table>

Table C.1 Decision tree of store id 11218 in conditions form. (Cont.)

Root Node (L1)	L2	L3	L4	L5	L6	L7
Pub_id = 86						
	Price <= 255					
		Year <= 2: low (4.0/2.0)				
		Year > 2: veryhigh (4.0/2.0)				
	Price > 255: medium (3.0/2.0)					
Pub_id = 87						
	Price <= 360: veryhigh (5.0/3.0)					
	Price > 360: medium (7.0/4.0)					
Pub_id = 88: high (2.0/1.0)						
Pub_id = 89: low (4.0/2.0)						
Pub_id = 91						
	Cat_id = 1					
		Price <= 255				
			Year <= 0: veryhigh (2.0/2.0)			
			Year > 0: high (3.0/2.0)			
		Price > 255: no (2.0/1.0)				
	Cat_id = 8: low (2.0/1.0)					
Pub_id = 92						
	Year <= 1: medium (4.0/2.0)					
	Year > 1					
		Year <= 2: veryhigh (3.0/2.0)				
		Year > 2: high (2.0/1.0)				
Pub_id = 93: high (6.0/2.0)						
Pub_id = 94						
	Price <= 145: high (3.0/2.0)					
	Price > 145: low (3.0/1.0)					
Pub_id = 95						
	Price <= 110: medium (3.0/2.0)					
	Price > 110					
		Price <= 155				
			Price <= 135: low (3.0/2.0)			
			Price > 135: high (6.0/3.0)			
		Price > 155: medium (2.0/1.0)				
Pub_id = 98						
	Year <= 0: medium (3.0/3.0)					
	Year > 0: veryhigh (3.0/2.0)					
Pub_id = 100: no (4.0/3.0)						
Pub_id = 103: no (3.0/3.0)						
Pub_id = 107: medium (2.0/2.0)						
Pub_id = 110						
	Year <= 7: medium (3.0/3.0)					
	Year > 7: low (2.0/1.0)					
Pub_id = 111						
	Price <= 155					
		Year <= 0: medium (6.0/4.0)				
		Year > 0				
			Year <= 8			
				Price <= 140: high (10.0/6.0)		
				Price > 140: low (2.0/1.0)		
			Year > 8: low (2.0/1.0)			
	Price > 155: low (9.0/5.0)					

Table C.1 Decision tree of store id 11218 in conditions form. (Cont.)

Root Node (L1)	L2	L3	L4	L5	L6	L7
Pub_id = 112: low (3.0/1.0)						
Pub_id = 115: low (2.0/1.0)						
Pub_id = 118						
	Year <= 0: no (2.0/2.0)					
	Year > 0: high (4.0/2.0)					
Pub_id = 123						
	Year <= 3: medium (2.0/1.0)					
	Year > 3: high (4.0/3.0)					
Pub_id = 125: high (4.0/3.0)						
Pub_id = 127: medium (2.0/1.0)						
Pub_id = 130: veryhigh (6.0/4.0)						
Pub_id = 137: no (3.0/2.0)						
Pub_id = 144						
	Year <= 7: low (2.0/1.0)					
	Year > 7: veryhigh (3.0/3.0)					
Pub_id = 150: low (2.0/1.0)						
Pub_id = 152						
	Year <= 2: veryhigh (3.0/2.0)					
	Year > 2: low (3.0/2.0)					
Pub_id = 153						
	Price <= 180: veryhigh (3.0/3.0)					
	Price > 180: low (6.0/3.0)					
Pub_id = 159: no (2.0/1.0)						
Pub_id = 161						
	Price <= 100: low (2.0/1.0)					
	Price > 100: medium (2.0/1.0)					
Pub_id = 170: no (1.0/1.0)						
Pub_id = 177: no (1.0/1.0)						
Pub_id = 184: high (2.0/1.0)						
Pub_id = 185: no (1.0/1.0)						
Pub_id = 189: no (1.0/1.0)						
Pub_id = 193: no (1.0/1.0)						
Pub_id = 198: no (1.0/1.0)						
Pub_id = 201: high (3.0/1.0)						
Pub_id = 216: no (1.0/1.0)						
Pub_id = 222: no (1.0/1.0)						
Pub_id = 223						
	Year <= 0: low (3.0/2.0)					
	Year > 0					
		Year <= 1: veryhigh (3.0/3.0)				
		Year > 1: high (3.0/2.0)				
Pub_id = 225: no (1.0/1.0)						
Pub_id = 246: no (1.0/1.0)						
Pub_id = 253: no (1.0/1.0)						
Pub_id = 262: no (1.0/1.0)						
Pub_id = 267: low (3.0/1.0)						
Pub_id = 272: no (1.0/1.0)						
Pub_id = 273: no (1.0/1.0)						

2.2. Decision Tree of Store ID 30121

Table C.2 Decision tree of store id 30121 in conditions form.

Root Node (L1)	L2	L3	L4
Pub_id = 1			
	Year <= 1		
		Year <= 0: low (6.0/2.0)	
		Year > 0	
			Price <= 125: no (13.0/9.0)
			Price > 125: high (2.0/1.0)
	Year > 1: medium (2.0/2.0)		
Pub_id = 3: low (2.0/1.0)			
Pub_id = 4: veryhigh (1.0/1.0)			
Pub_id = 7: medium (2.0/1.0)			
Pub_id = 8: veryhigh (5.0/4.0)			
Pub_id = 12: low (3.0/1.0)			
Pub_id = 13: low (3.0/1.0)			
Pub_id = 14: veryhigh (2.0/2.0)			
Pub_id = 15: no (2.0/1.0)			
Pub_id = 17: high (1.0/1.0)			
Pub_id = 19: low (2.0/1.0)			
Pub_id = 20: veryhigh (1.0/1.0)			
Pub_id = 21: veryhigh (18.0/11.0)			
Pub_id = 22: low (2.0/1.0)			
Pub_id = 23			
	Price <= 450: veryhigh (5.0/3.0)		
	Price > 450: low (2.0/1.0)		
Pub_id = 24: medium (13.0/4.0)			
Pub_id = 25			
	Cat_id = 6: medium (2.0/2.0)		
	Cat_id = 7: low (0.0)		
	Cat_id = 8: low (0.0)		
	Cat_id = 9: low (3.0/2.0)		
Pub_id = 26: medium (2.0/1.0)			
Pub_id = 28: no (2.0/1.0)			
Pub_id = 29: no (4.0/4.0)			
Pub_id = 30: low (2.0/1.0)			
Pub_id = 31: medium (1.0/1.0)			
Pub_id = 33: low (2.0/1.0)			
Pub_id = 34: no (1.0/1.0)			
Pub_id = 36: veryhigh (3.0/2.0)			
Pub_id = 41: medium (4.0/4.0)			
Pub_id = 42			
	Cat_id = 1: low (3.0/2.0)		
	Cat_id = 12: medium (2.0/1.0)		
Pub_id = 44: no (1.0/1.0)			
Pub_id = 47			
	Price <= 95		
		Year <= 3: high (3.0/2.0)	
		Year > 3: low (6.0/3.0)	
	Price > 95: medium (6.0/3.0)		
Pub_id = 48: high (3.0/2.0)			
Pub_id = 49: no (1.0/1.0)			
Pub_id = 50: no (1.0/1.0)			

Table C.2 Decision tree of store id 30121 in conditions form (Cont.)

Root Node (L1)	L2	L3	L4
Pub_id = 54			
	Year <= 1		
		Cat_id = 4: high (4.0/2.0)	
		Cat_id = 11: medium (8.0/4.0)	
	Year > 1: low (6.0/5.0)		
Pub_id = 56: medium (4.0/2.0)			
Pub_id = 60: no (3.0/1.0)			
Pub_id = 61: no (3.0/1.0)			
Pub_id = 62: high (1.0/1.0)			
Pub_id = 63			
	Year <= 0		
		Price <= 170: low (9.0/7.0)	
		Price > 170: medium (5.0/3.0)	
	Year > 0		
		Price <= 180: no (17.0/13.0)	
		Price > 180: medium (3.0/2.0)	
Pub_id = 64: no (53.0/30.0)			
Pub_id = 65			
	Year <= 0: high (5.0/3.0)		
	Year > 0: medium (5.0/2.0)		
Pub_id = 66: high (28.0/13.0)			
Pub_id = 67: low (2.0/1.0)			
Pub_id = 68			
	Year <= 2: no (6.0/2.0)		
	Year > 2: veryhigh (4.0/4.0)		
Pub_id = 69			
	Year <= 0: low (6.0/4.0)		
	Year > 0		
		Cat_id = 2: medium (4.0/2.0)	
		Cat_id = 31: no (3.0/2.0)	
Pub_id = 70: veryhigh (1.0/1.0)			
Pub_id = 71			
	Cat_id = 7: low (2.0/1.0)		
	Cat_id = 11: veryhigh (3.0/2.0)		
Pub_id = 72: medium (6.0/4.0)			
Pub_id = 74: medium (2.0/2.0)			
Pub_id = 80: low (4.0/2.0)			
Pub_id = 81: low (3.0/1.0)			
Pub_id = 82: low (8.0/3.0)			
Pub_id = 83: veryhigh (4.0/3.0)			
Pub_id = 84: high (4.0/2.0)			
Pub_id = 85: no (1.0/1.0)			
Pub_id = 86: no (7.0/2.0)			
Pub_id = 87: high (16.0/6.0)			
Pub_id = 89: veryhigh (1.0/1.0)			
Pub_id = 91			
	Cat_id = 1		
		Year <= 0: high (2.0/1.0)	
		Year > 0: no (2.0/1.0)	
	Cat_id = 6: low (3.0/1.0)		
	Cat_id = 7: veryhigh (3.0/3.0)		
	Cat_id = 8		
		Price <= 280: medium (3.0/2.0)	
		Price > 280: high (2.0/2.0)	
	Cat_id = 21: low (2.0/1.0)		

Table C.2 Decision tree of store id 30121 in conditions form. (Cont.)

Root Node (L1)	L2	L3	L4
Pub_id = 92			
	Year <= 1: high (2.0/1.0)		
	Year > 1: medium (3.0/2.0)		
Pub_id = 93			
	Cat_id = 2: no (2.0/1.0)		
	Cat_id = 11: veryhigh (3.0/3.0)		
Pub_id = 94			
	Price <= 149: medium (3.0/2.0)		
	Price > 149: low (2.0/1.0)		
Pub_id = 98			
	Year <= 1: veryhigh (5.0/5.0)		
	Year > 1: low (2.0/1.0)		
Pub_id = 100: low (2.0/1.0)			
Pub_id = 102			
	Cat_id = 1: low (2.0/1.0)		
	Cat_id = 6: no (2.0/1.0)		
Pub_id = 103: no (1.0/1.0)			
Pub_id = 104: no (1.0/1.0)			
Pub_id = 105: no (3.0/3.0)			
Pub_id = 107: veryhigh (3.0/2.0)			
Pub_id = 110: medium (5.0/3.0)			
Pub_id = 111			
	Price <= 280: no (2.0/2.0)		
	Price > 280: low (4.0/2.0)		
Pub_id = 112: high (1.0/1.0)			
Pub_id = 113: high (1.0/1.0)			
Pub_id = 118: no (2.0/1.0)			
Pub_id = 119: high (2.0/1.0)			
Pub_id = 123: high (3.0/2.0)			
Pub_id = 124: high (3.0/2.0)			
Pub_id = 126: veryhigh (2.0/2.0)			
Pub_id = 130			
	Year <= 5: medium (3.0/3.0)		
	Year > 5: veryhigh (2.0/3.0)		
Pub_id = 132: no (2.0/1.0)			
Pub_id = 136			
	Price <= 150: veryhigh (3.0/2.0)		
	Price > 150: high (4.0/3.0)		
Pub_id = 144: medium (3.0/2.0)			
Pub_id = 146: veryhigh (1.0/1.0)			
Pub_id = 152: medium (2.0/2.0)			
Pub_id = 153: veryhigh (2.0/2.0)			
Pub_id = 162: medium (2.0/1.0)			
Pub_id = 164: veryhigh (1.0/1.0)			
Pub_id = 190: high (1.0/1.0)			
Pub_id = 194: medium (4.0/3.0)			
Pub_id = 196: low (2.0/1.0)			
Pub_id = 204: no (2.0/1.0)			
Pub_id = 219: low (2.0/1.0)			
Pub_id = 223			
	Year <= 1: high (2.0/1.0)		
	Year > 1: low (2.0/2.0)		
Pub_id = 250: no (2.0/1.0)			

2.3 Decision Tree of Store ID 83205.

Table C.3 Decision tree of store id 83205 in conditions form.

Root Node (L1)	L2	L3	L4	L5
Pub_id = 1				
	Year <= 0			
		Price <= 100: high (5.0/4.0)		
		Price > 100: medium (2.0/1.0)		
	Year > 0			
		Price <= 125: low (2.0/2.0)		
		Price > 125: medium (2.0/1.0)		
Pub_id = 3: high (2.0/1.0)				
Pub_id = 8				
	Year <= 1			
		Price <= 70: high (3.0/3.0)		
		Price > 70: veryhigh (3.0/2.0)		
	Year > 1: low (2.0/2.0)			
Pub_id = 13				
	Cat_id = 8: low (2.0/1.0)			
	Cat_id = 21: high (3.0)			
Pub_id = 14: low (2.0/1.0)				
Pub_id = 16: high (1.0/1.0)				
Pub_id = 22				
	Price <= 140: low (7.0/4.0)			
	Price > 140: medium (2.0/2.0)			
Pub_id = 23: low (2.0/1.0)				
Pub_id = 24: low (9.0/7.0)				
Pub_id = 25: high (3.0/2.0)				
Pub_id = 26				
	Year <= 0: high (4.0/2.0)			
	Year > 0: low (4.0/4.0)			
Pub_id = 29: high (1.0/1.0)				
Pub_id = 31				
	Year <= 3: high (8.0/6.0)			
	Year > 3			
		Year <= 4: low (3.0/2.0)		
		Year > 4: high (3.0/2.0)		
Pub_id = 33: low (2.0/1.0)				
Pub_id = 34: medium (4.0/3.0)				
Pub_id = 36: medium (1.0/1.0)				
Pub_id = 37: low (2.0/1.0)				
Pub_id = 38: low (3.0/2.0)				
Pub_id = 39: high (32.0/18.0)				
Pub_id = 41				
	Price <= 130: low (7.0/3.0)			
	Price > 130: veryhigh (9.0/7.0)			
Pub_id = 42: low (3.0/1.0)				
Pub_id = 43: low (2.0/1.0)				
Pub_id = 45: low (2.0/1.0)				
Pub_id = 47				
	Price <= 110: low (2.0/1.0)			
	Price > 110			
		Cat_id = 1: veryhigh (3.0/2.0)		
		Cat_id = 4: medium (2.0)		
Pub_id = 48: low (11.0/7.0)				
Pub_id = 49				
	Cat_id = 0: medium (2.0/1.0)			
	Cat_id = 1: low (2.0/1.0)			
Pub_id = 51: medium (1.0/1.0)				
Pub_id = 52: low (2.0/2.0)				

Table C.3 Decision tree of store id 83205 in conditions form. (Cont.)

Root Node (L1)	L2	L3	L4	L5
Pub_id = 54				
	| Year <= 0: veryhigh (26.0/15.0)			
	| Year > 0			
		| | Year <= 1		
			| | | Price <= 40: low (10.0/5.0)	
			| | | Price > 40: veryhigh (9.0/5.0)	
		| | Year > 1: low (11.0/7.0)		
Pub_id = 56: low (6.0/4.0)				
Pub_id = 63				
	| Year <= 4			
		| | Year <= 0: medium (8.0/5.0)		
		| | Year > 0		
			| | | Year <= 3: low (7.0/5.0)	
			| | | Year > 3: medium (3.0/2.0)	
	| Year > 4			
		| | Price <= 170		
			| | | Year <= 5: veryhigh (3.0/2.0)	
			| | | Year > 5: low (2.0/2.0)	
		| | Price > 170: high (3.0/3.0)		
Pub_id = 64				
	| Year <= 0: medium (48.0/27.0)			
	| Year > 0			
		| | Year <= 1		
			| | | Price <= 230: low (8.0/7.0)	
			| | | Price > 230	
				| | | | Price <= 250: medium (6.0/4.0)
				| | | | Price > 250: low (19.0/11.0)
		| | Year > 1: low (5.0/5.0)		
Pub_id = 65: high (1.0/1.0)				
Pub_id = 66: low (3.0/1.0)				
Pub_id = 67: medium (4.0/2.0)				
Pub_id = 68: low (6.0/3.0)				
Pub_id = 69: low (31.0/9.0)				
Pub_id = 72: low (2.0/1.0)				
Pub_id = 74: high (1.0/1.0)				
Pub_id = 78				
	| Year <= 0: medium (3.0/2.0)			
	| Year > 0			
		| | Year <= 1: veryhigh (3.0/2.0)		
		| | Year > 1: low (3.0/2.0)		
Pub_id = 82: high (1.0/1.0)				
Pub_id = 83: veryhigh (3.0/2.0)				
Pub_id = 86: high (1.0/1.0)				
Pub_id = 87				
	| Price <= 125: low (2.0/2.0)			
	| Price > 125: medium (5.0/2.0)			
Pub_id = 91: low (5.0/3.0)				
Pub_id = 92: low (2.0/1.0)				
Pub_id = 93: low (2.0/1.0)				
Pub_id = 95				
	| Price <= 110: medium (6.0/5.0)			
	| Price > 110			
		| | Year <= 2: low (3.0/2.0)		
		| | Year > 2: high (5.0/3.0)		

Table C.3 Decision tree of store id 83205 in conditions form. (Cont.)

Root Node (L1)	L2	L3	L4	L5
Pub_id = 98: low (2.0/1.0)				
Pub_id = 99: low (7.0/4.0)				
Pub_id = 110				
	| Price <= 100: low (3.0/2.0)			
	| Price > 100: medium (2.0/1.0)			
Pub_id = 111: low (2.0/1.0)				
Pub_id = 115: high (1.0/1.0)				
Pub_id = 126: veryhigh (2.0/2.0)				
Pub_id = 130: low (6.0/3.0)				
Pub_id = 132: low (5.0/2.0)				
Pub_id = 133: veryhigh (1.0/1.0)				
Pub_id = 134: medium (3.0/1.0)				
Pub_id = 141: low (2.0/1.0)				
Pub_id = 142: medium (3.0/1.0)				
Pub_id = 144: low (2.0/1.0)				
Pub_id = 151: low (2.0/1.0)				
Pub_id = 155				
	| Year <= 6: veryhigh (3.0/2.0)			
	| Year > 6: high (3.0/2.0)			
Pub_id = 190: low (2.0/1.0)				
Pub_id = 191: high (1.0/1.0)				
Pub_id = 194: low (2.0/1.0)				
Pub_id = 196: low (2.0/1.0)				
Pub_id = 226: low (3.0/1.0)				
Pub_id = 266: high (3.0/2.0)				
Pub_id = 267: high (5.0/3.0)				

BIOGRAPHY

NAME	Miss. Patthama Saeting
DATE OF BIRTH	2 May 1986
PLACE OF BIRTH	Chumphon, Thailand
INSTITUTIONS ATTENDED	Taksin University, 2004-2008 Bachelor of Computer Science Mahidol University, 2009-2013 Master of Science (Technology of Information System Management)
HOME ADDRESS	87/490 Soi 30, Passorn 7 village, Moo 6 Rd. Bang kruai - Sai noi Rd, Bang ruk yai, Bang Bua Thong, Nonthaburi 11110 Tel. 081-856-2529 E-mail : pat_hugo@hotmail.com
PUBLICATION / PRESENTATION	Patthama Saeting, A comparative study of book distribution techniques for maximizing sell benefit, The 2012 International Computer Science and Engineering Conference, October 18-19, 2012, Pattaya, Thailand