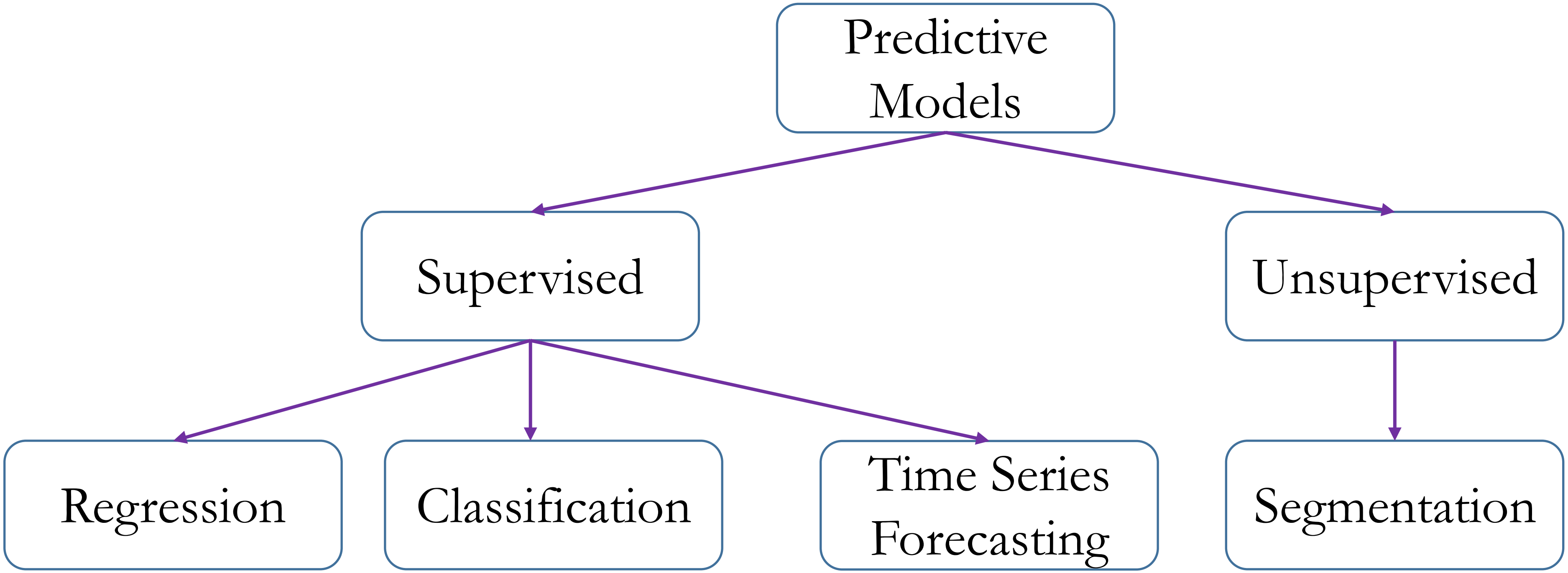
Trees

Previous class

- Regression-based Forecasting



Predictive Models

Supervised

Unsupervised

Regression

Classification

Time Series Forecasting

Segmentation

- ***k*-Nearest Neighbor**
- **Linear Regression**
- **Regression Trees**
- Neural Networks
- Ensembles
- ……

- ***k*-Nearest Neighbor**
- Naïve Bayes
- **Logistic Regression**
- **Classification Trees**
- Neural Networks
- Discriminant Analysis
- Ensembles
- ……

- **Regression-based**
- Smoothing methods
- ……

- **Clustering**
- ……

Trees

- Flexible data-driven method
- Used for
  - Classification (called Classification Tree)
  - Regression (called Regression Tree)
- Transparent
- Easy interpretation
- Doesn't require enormous effort
- Method
  - **Recursive Partitioning** : Separating records into subgroups by creating splits on predictors

Recursive Partitioning

- Outcome variable Y
- Predictor variables $X_1, X_2, X_3, \cdots\cdots X_p$
- Recursive Partitioning
    - Divides the p-dimensional space of predictors into non-overlapping multidimensional rectangles
- Accomplished recursively
    - Operating on the results of prior division
- Idea is to divide the entire variable-space up into rectangles such that each rectangle is as **homogeneous** or **pure**
- **Homogeneous** or **Pure** meaning containing records mostly of one class

Data on Riding Mowers

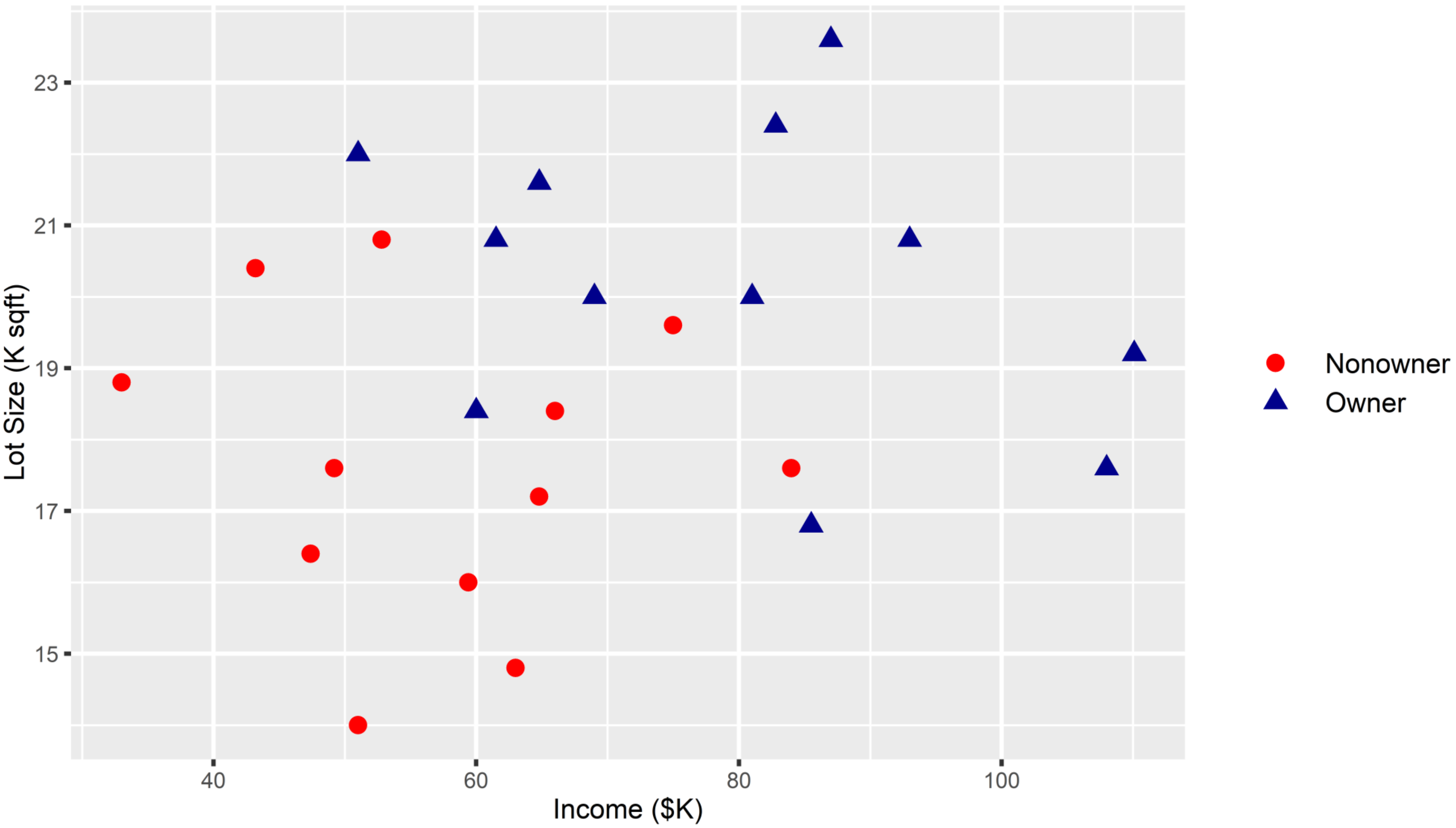
- Riding-mower manufacturer would like to find a way of classifying families in a city into an **owner** or **non-owner**
- Attributes
  - Income : Income of the household in thousand of dollars
  - Lot Size : Lot size in thousand of square foot
  - Ownership : Owner or Non-owner

Income	Lot_Size	Ownership
60	18.4	Owner
85.5	16.8	Owner
64.8	21.6	Owner
61.5	20.8	Owner

⋮

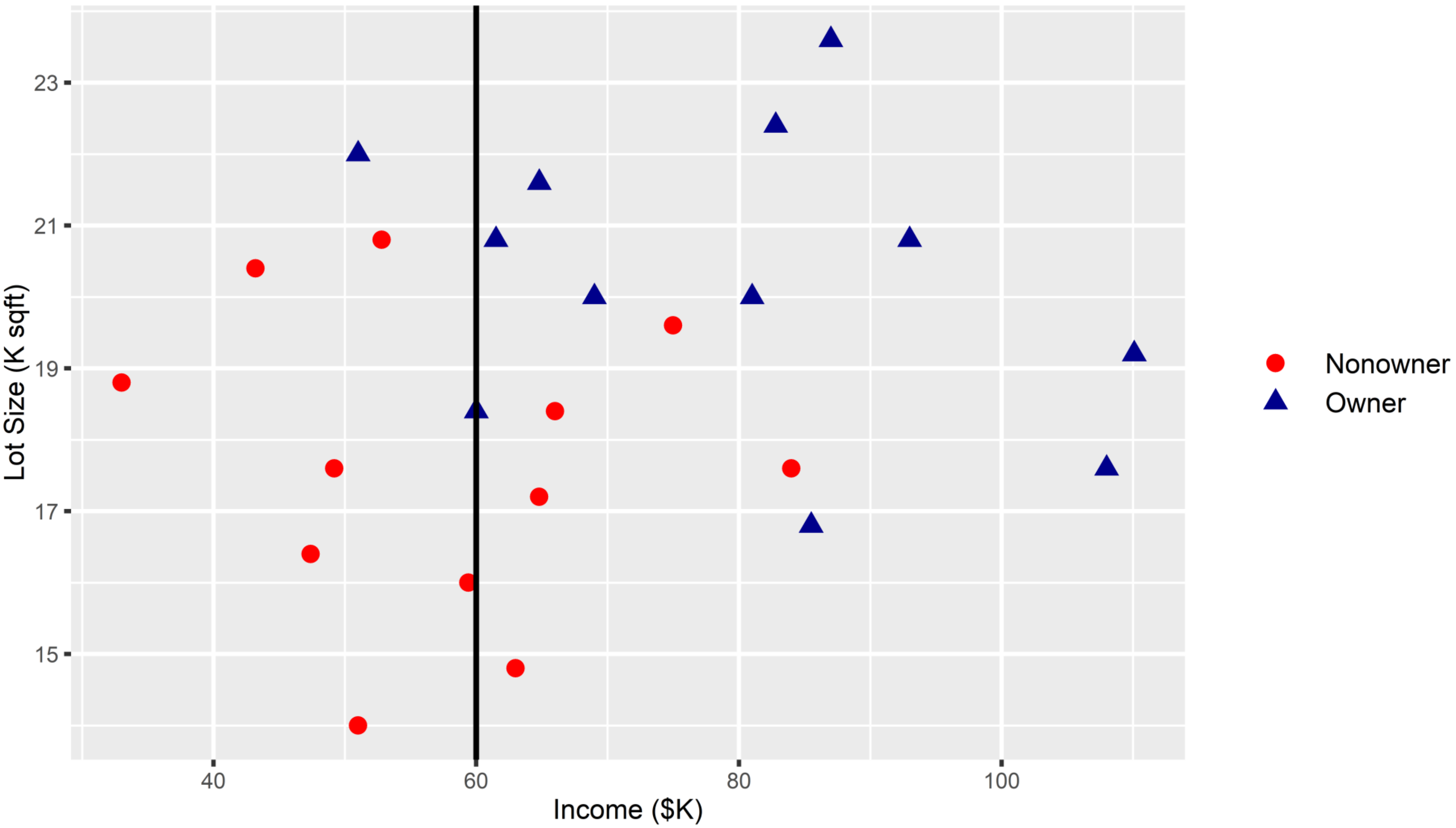
⋮

Scatter plot



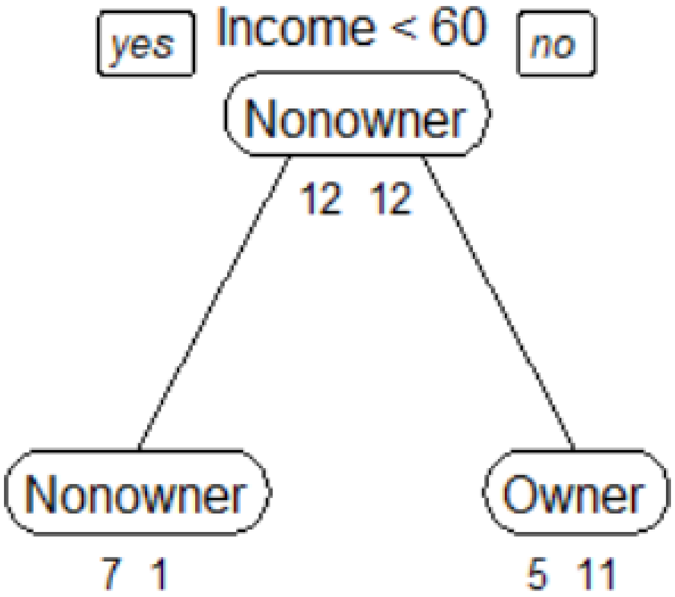
© Pradeep Pendem

First split at Income = 60

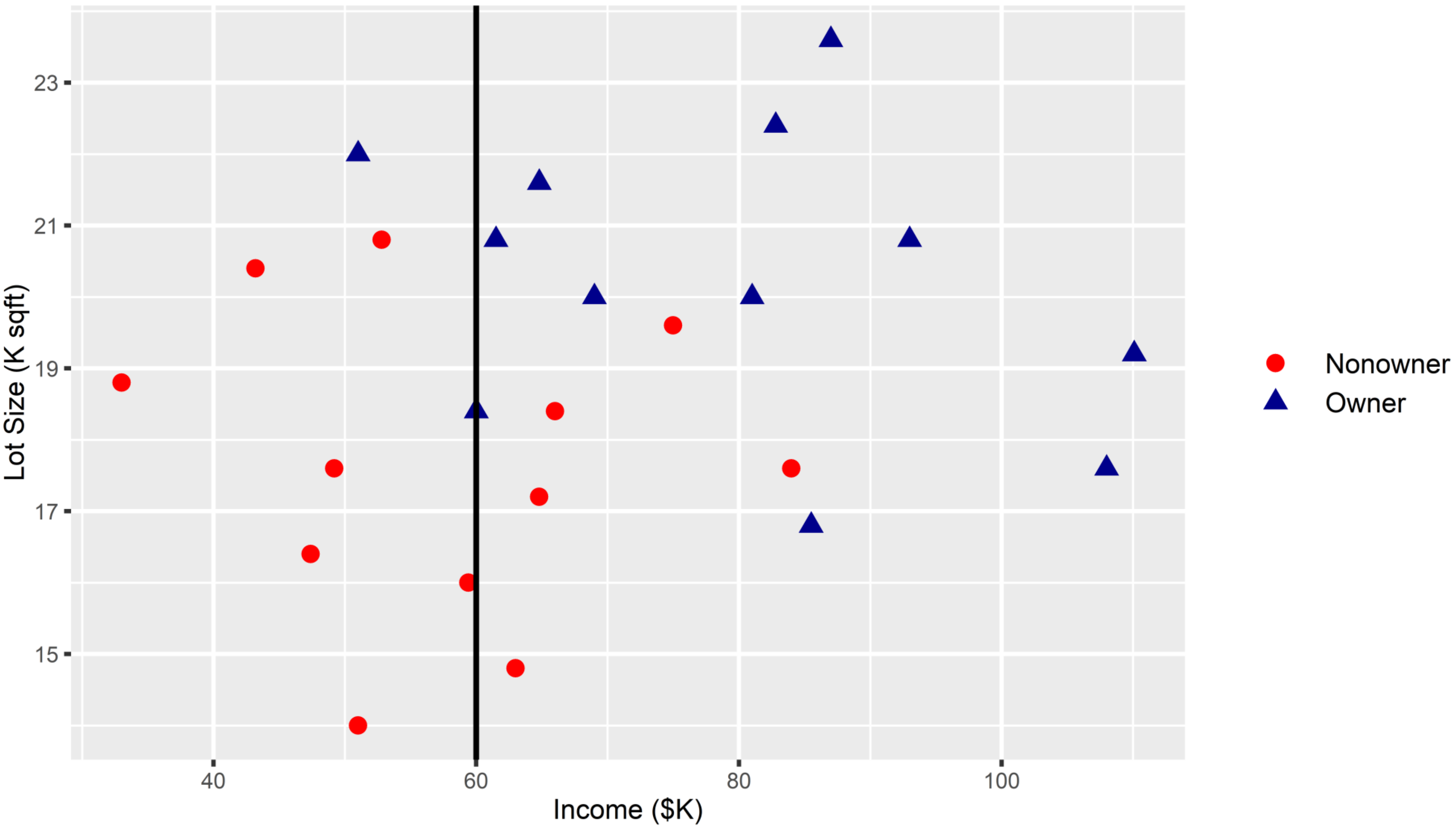


© Pradeep Pendem

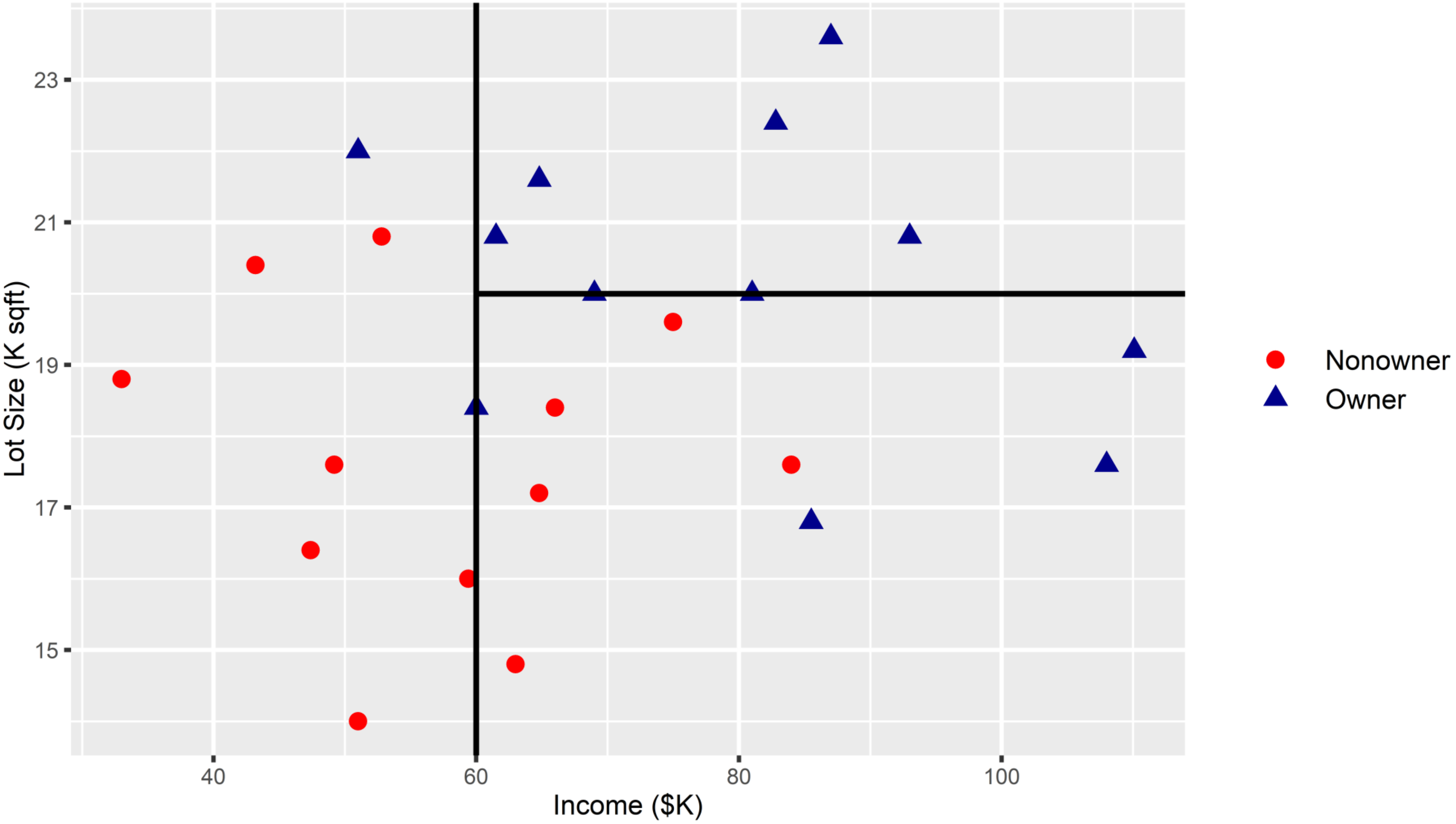
First split at Income = 60



First split at Income = 60

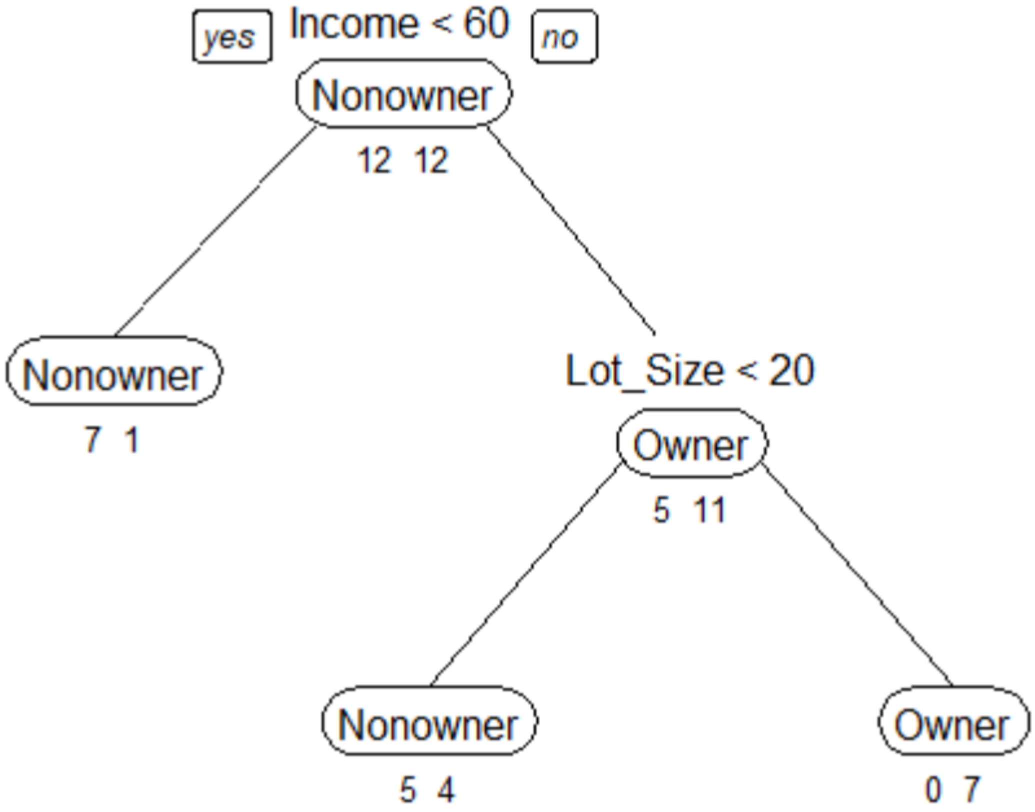


Second split at Lot Size = 20

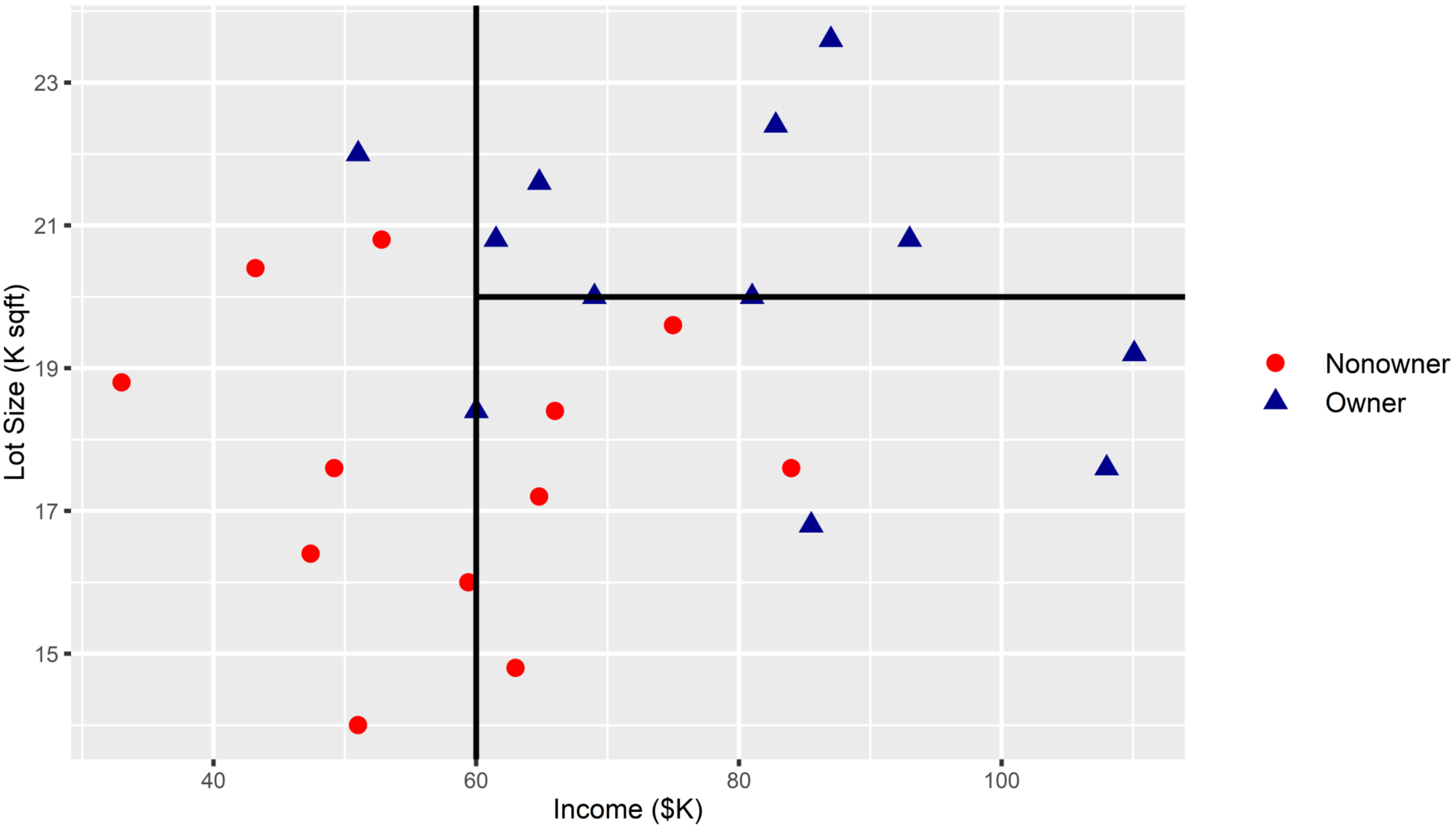


© Pradeep Pendem

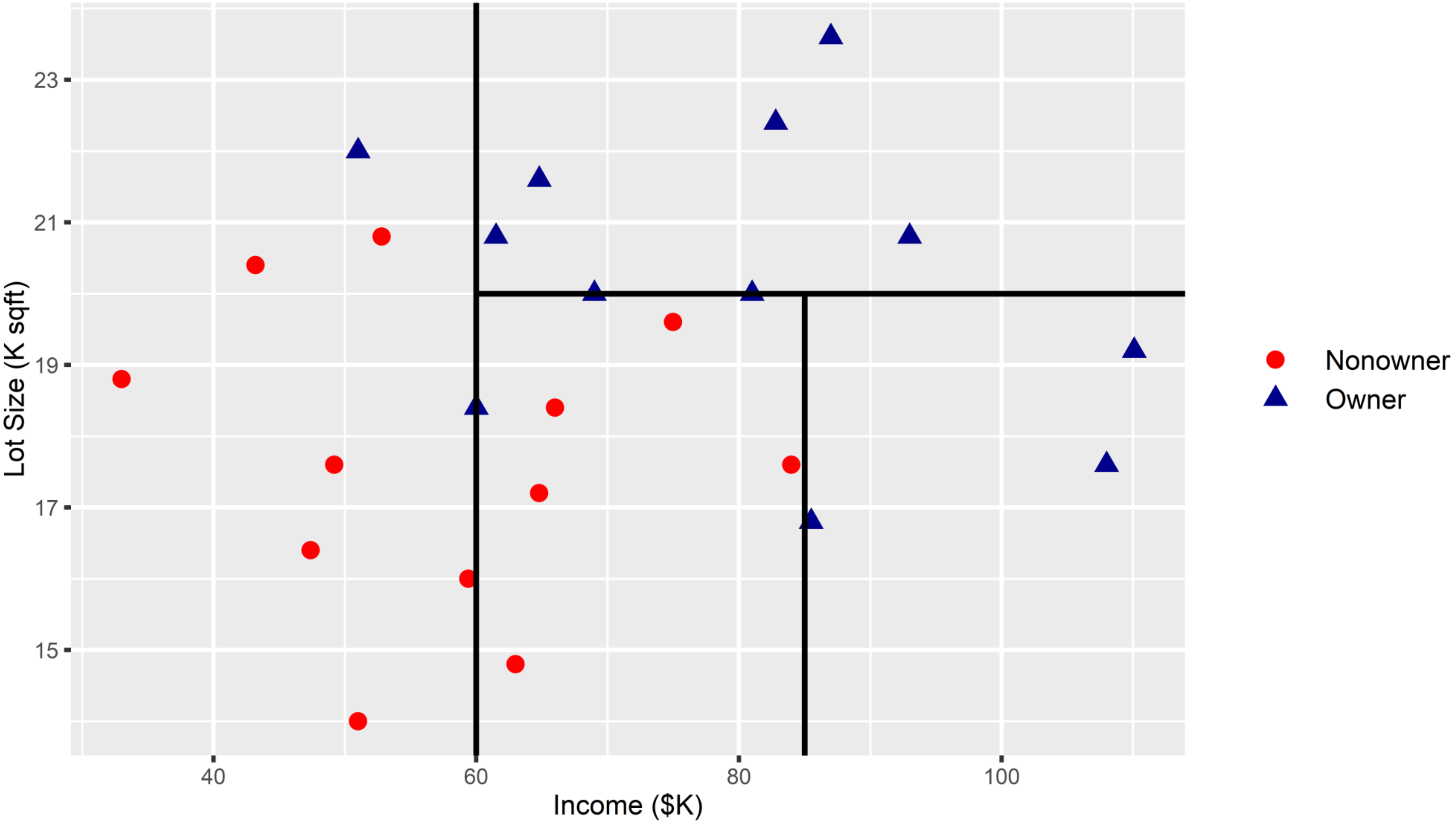
Second split at Lot Size = 20



Second split at Lot Size = 20

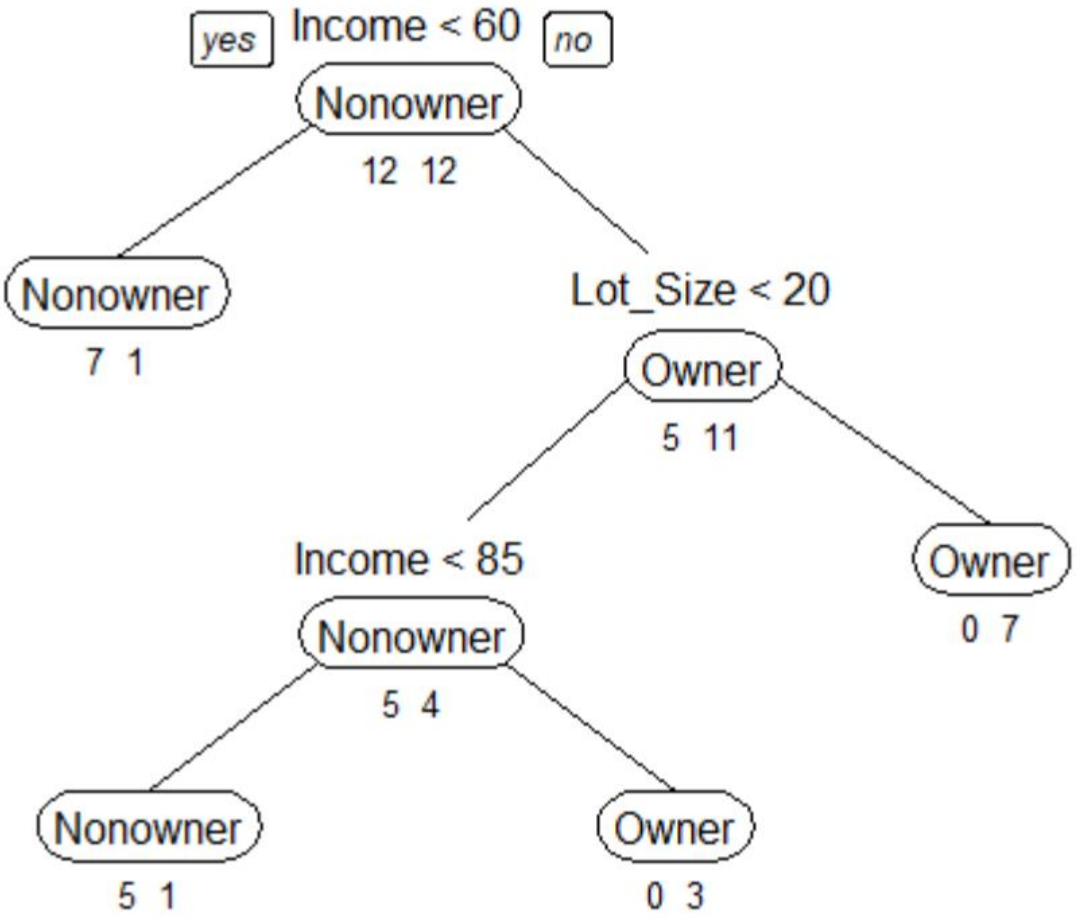


Third split at Income = 85



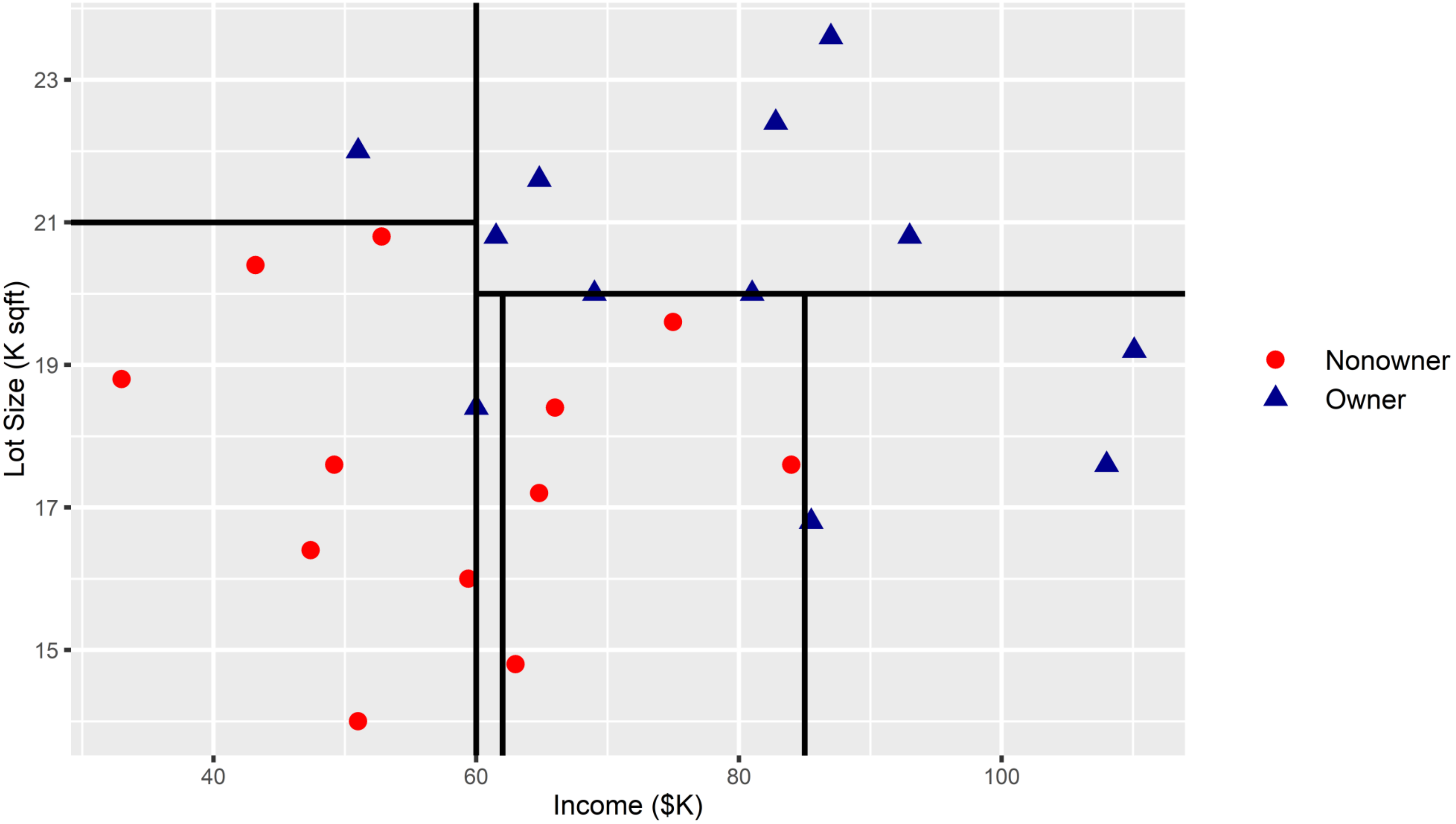
© Pradeep Pendem

Third split at Income = 85

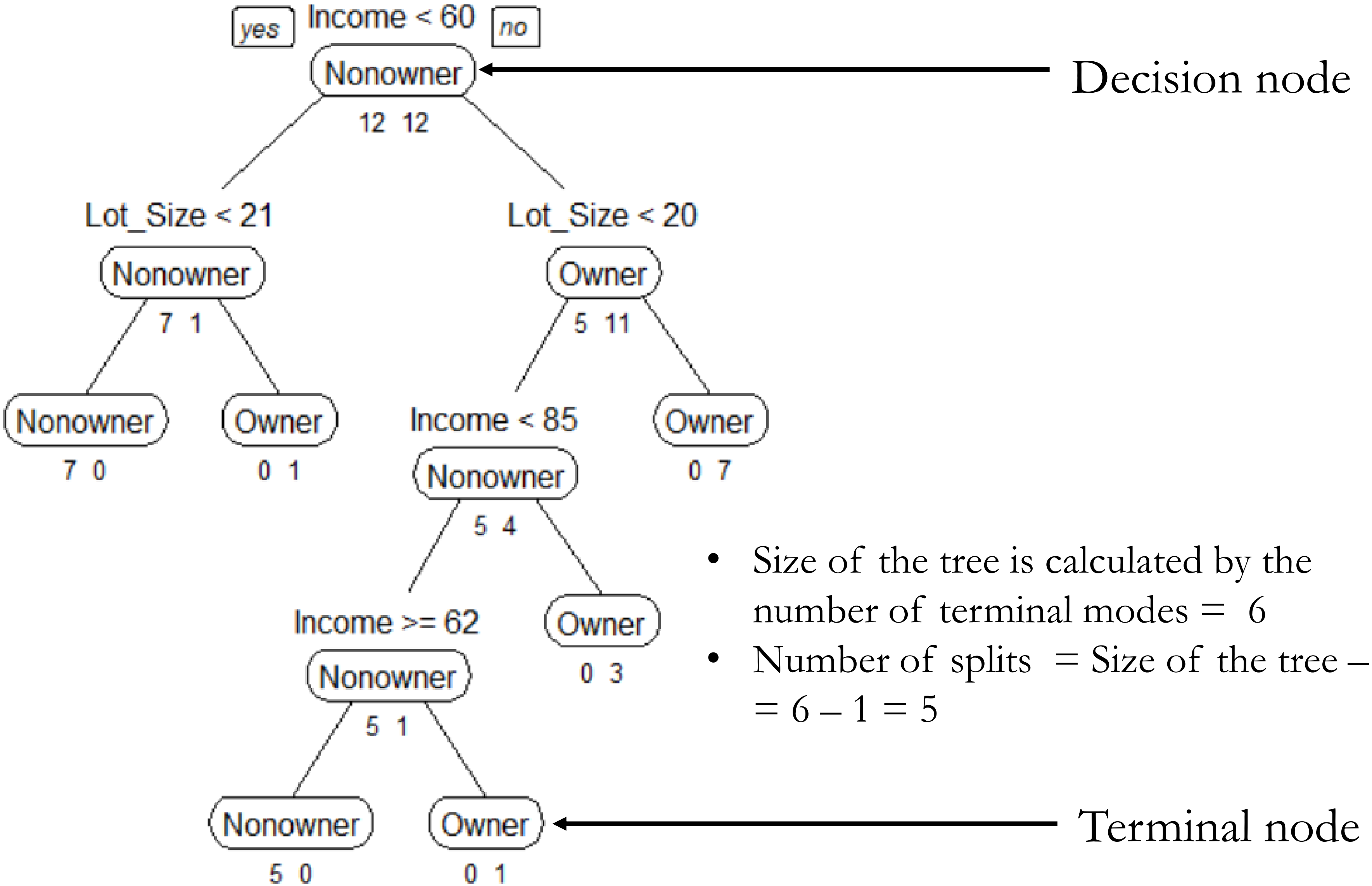


© Pradeep Pendem

Exhaustive splits



Fully grown tree



Decision node

Terminal node

- Size of the tree is calculated by the number of terminal modes = 6
- Number of splits = Size of the tree – 1 = 6 – 1 = 5

Measures of Impurity

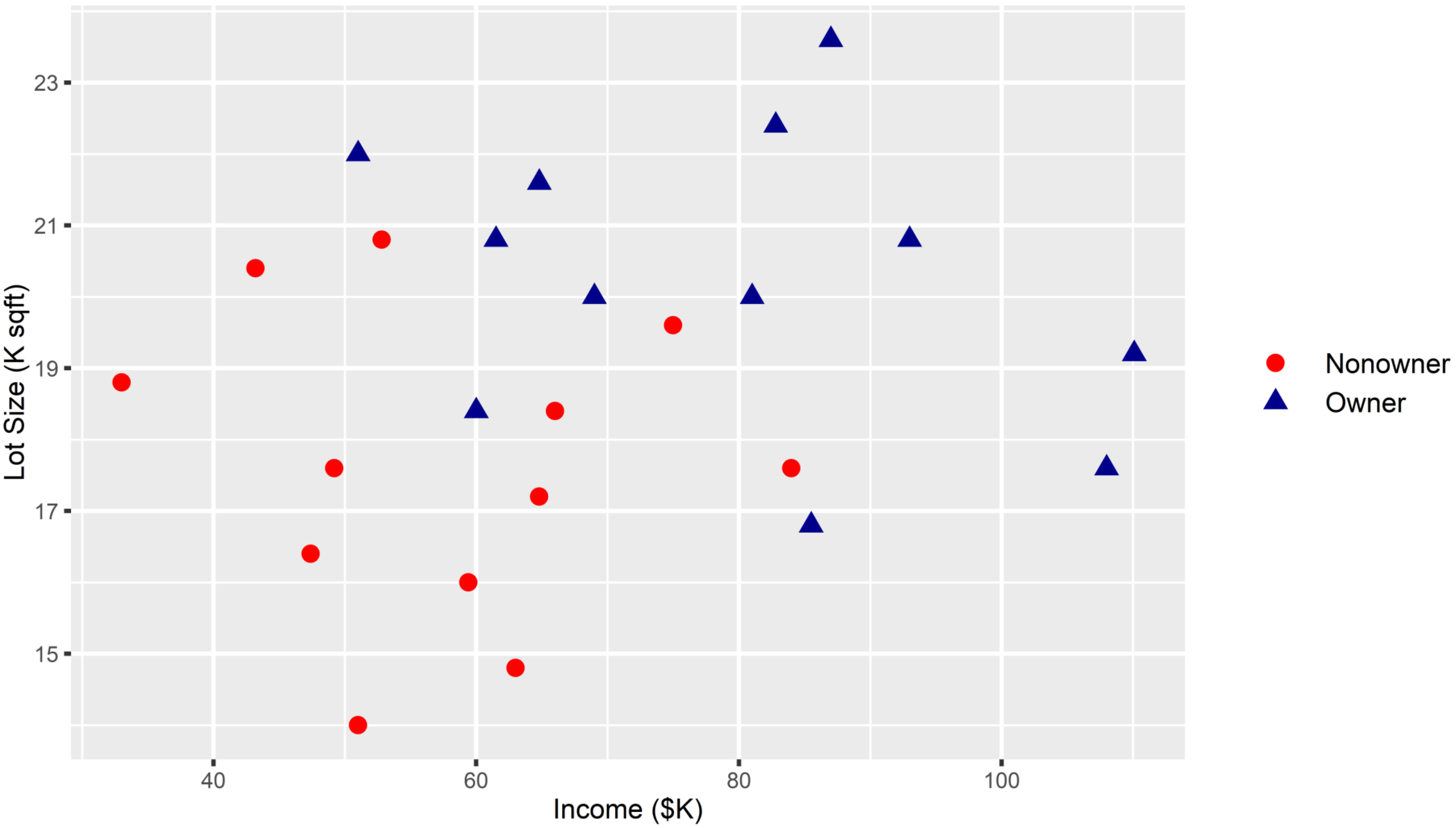
- Two popular measures
  - Gini Index
  - Entropy measure
- Gini impurity index for a rectangle A is given by

$$\mathrm{I(A)} = 1 - \sum_{\mathrm{k=1}}^{\mathrm{m}} \mathrm{p}_{\mathrm{k}}^{2}$$

- $\mathrm{p_k}$ is the proportion of records in rectangle A that belong to class k
- Measure takes values between 0 (when all records belong to same class) and ${}^{m-1}/_{m}$ (when all m classes are equally represented)

© Pradeep Pendem

Gini Index with no split



Gini Index : No-split

- Gini impurity index for no-split

$$= 1 - \sum_{k=1}^{m} p_k^2$$

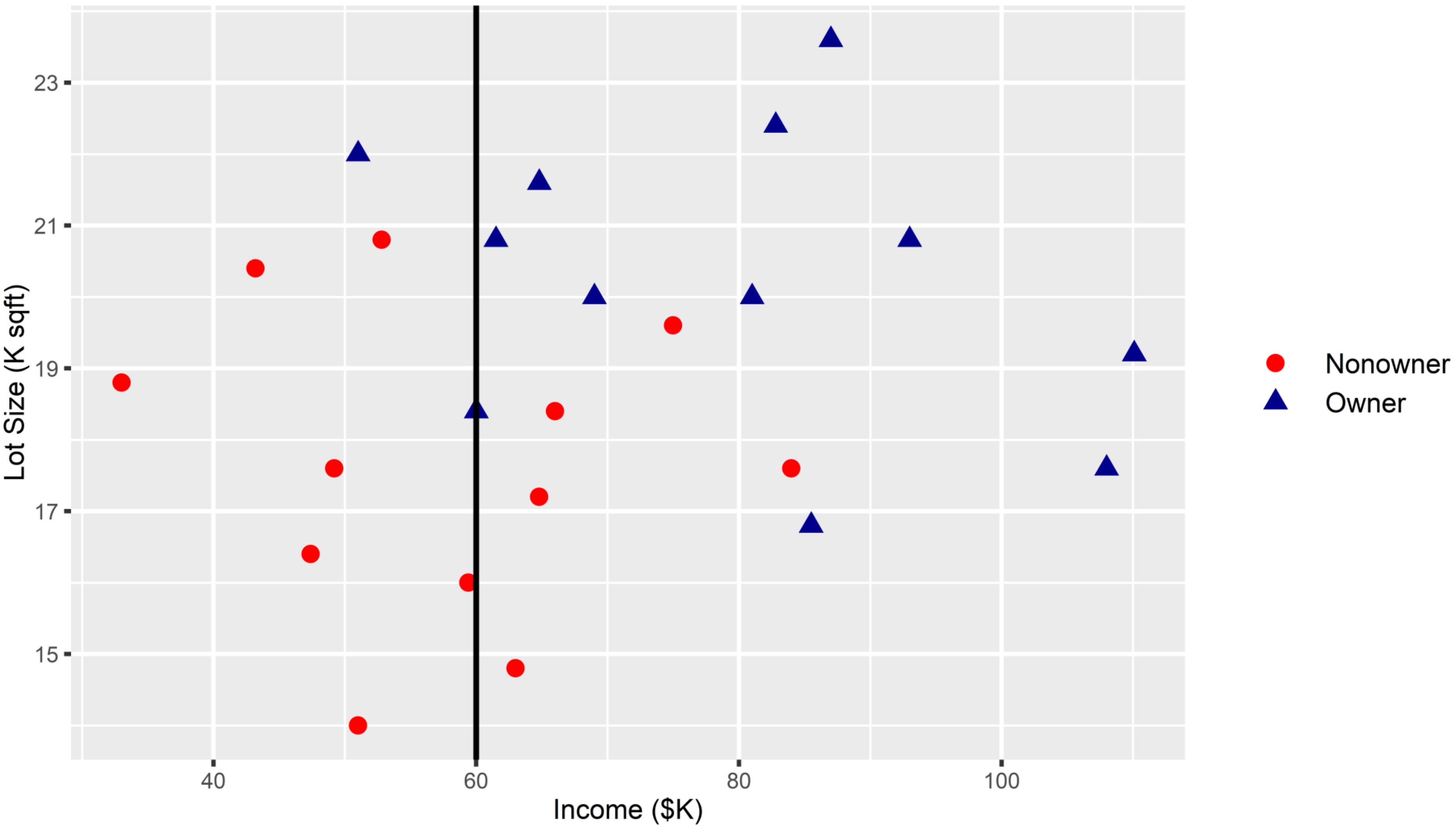
$$= 1 - \left(\frac{1}{2^2} + \frac{1}{2^2}\right) = ½$$

- p_k is the proportion of records in rectangle A that belong to class k

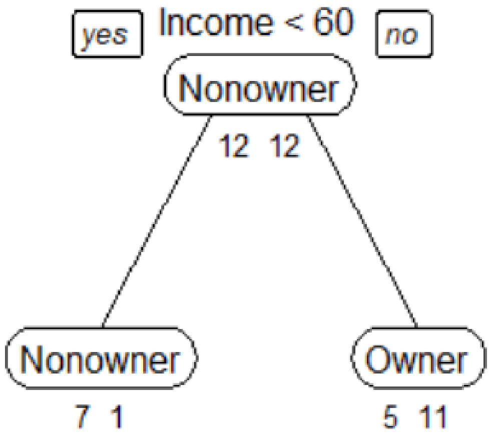
- Measure takes values between 0 (when all records belong to same class) and ${}^{m-1}/_{m}$ (when all m classes are equally represented)

© Pradeep Pendem

First split at Income = 60



Gini Index at First split of Income = 60



- Gini impurity index for <u>left</u> rectangle

$$= 1 - \sum_{k=1}^{m} p_k^2 \ = 1 - \left(\frac{7^2}{8^2} + \frac{1^2}{8^2}\right) = 0.219$$

- Gini impurity index for <u>right</u> rectangle

$$= 1 - \sum_{k=1}^{m} p_k^2 \ = 1 - \left(\frac{11^2}{16^2} + \frac{5^2}{16^2}\right) = 0.430$$

- Weighted Index $= \frac{8}{24} * 0.219 + \frac{16}{24} * 0.430 = 0.359$
- No split to first split → 0.5 to 0.359
- This is the lowest drop that can be expected, hence the choice 60

© Pradeep Pendem

Measures of Impurity

- Entropy

- Entropy for a rectangle A is given by

$$\text{entropy(A)} = -\sum_{\text{k=1}}^{\text{m}} p_k * \log_2 p_k$$

- $\mathrm{p_k}$ is the proportion of records in rectangle A that belong to class k

- Measure takes values between 0 (when all records belong to same class) and $\log_2(m)$(when all m classes are equally represented)

© Pradeep Pendem

Today’s class mandatory steps

- Create a folder name “**n. classification_tree**” within the folder

  “**oba_455_555_ddpm_r/rproject**”

- Download “**classification_tree_code.R**”, and all **csv** files from canvas
- Place all downloaded files in

  “**oba_455_555_ddpm_r /rproject/ n. classification_tree**”

- Open RStudio project
- Open “**classification_tree_code.R**” file within RStudio

Example : Acceptance of Personal Loan

- Response : Bank customer accepting a loan (1) or not (0)

- Predictors (X)

  - Age, Experience, Income, Family Size, Education
  - Spending on Credit cards
  - Mortgage, Securities account
  - Online banking
  - ……

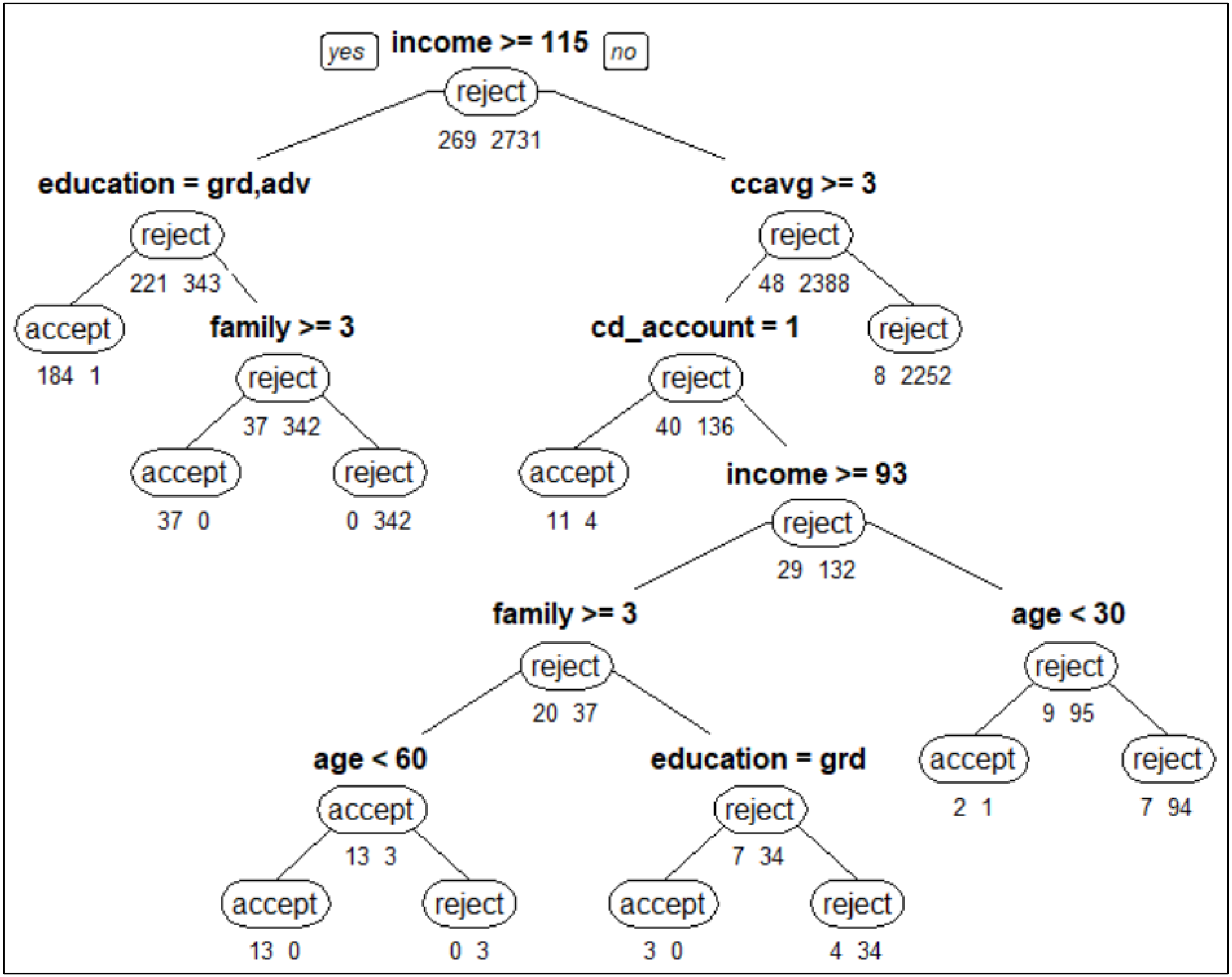
Pruning a tree

- Step 1: Set the seed, create train and validation data
- Step 2: Run a tree with options cp = 0.00001, minsplit = 5 or 10, xval = 5 or 10
- Step 3: Plot the cp or relative error
- Step 4: Find "nsplit" value and its associated cp value from “size of the tree”
- Step 5: Prune the tree with the optimal cp
- Step 6: Predict the loan status for validation data.
- Step 7 : Develop confusion matrix and accuracy measures

Pruning – Key variables

- Complexity parameter (cp)
    - Any split that does not improve the fit by cp is not attempted
    - Saves computing time by pruning off splits that are not worthwhile
- minsplit
    - minimum number of observations that must exist in a node in order for a split to be attempted.

Pruned tree


yes
income >= 115
no
reject
269 2731
education = grd,adv
reject
221 343
accept
184 1
family >= 3
reject
37 342
accept
37 0
reject
0 342
ccavg >= 3
reject
48 2388
cd_account = 1
reject
40 136
accept
11 4
income >= 93
reject
29 132
family >= 3
reject
20 37
age < 60
accept
13 3
accept
13 0
reject
0 3
education = grd
reject
7 34
accept
3 0
reject
4 34
age < 30
reject
9 95
accept
2 1
reject
7 94
reject
8 2252

Next class

- Regression Tree

Thank You