Regression Trees

Previous class & Announcements

- Classification Trees

Today's class

- Recap on Classification Tree
- Regression Tree

Recap of Classification Tree

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

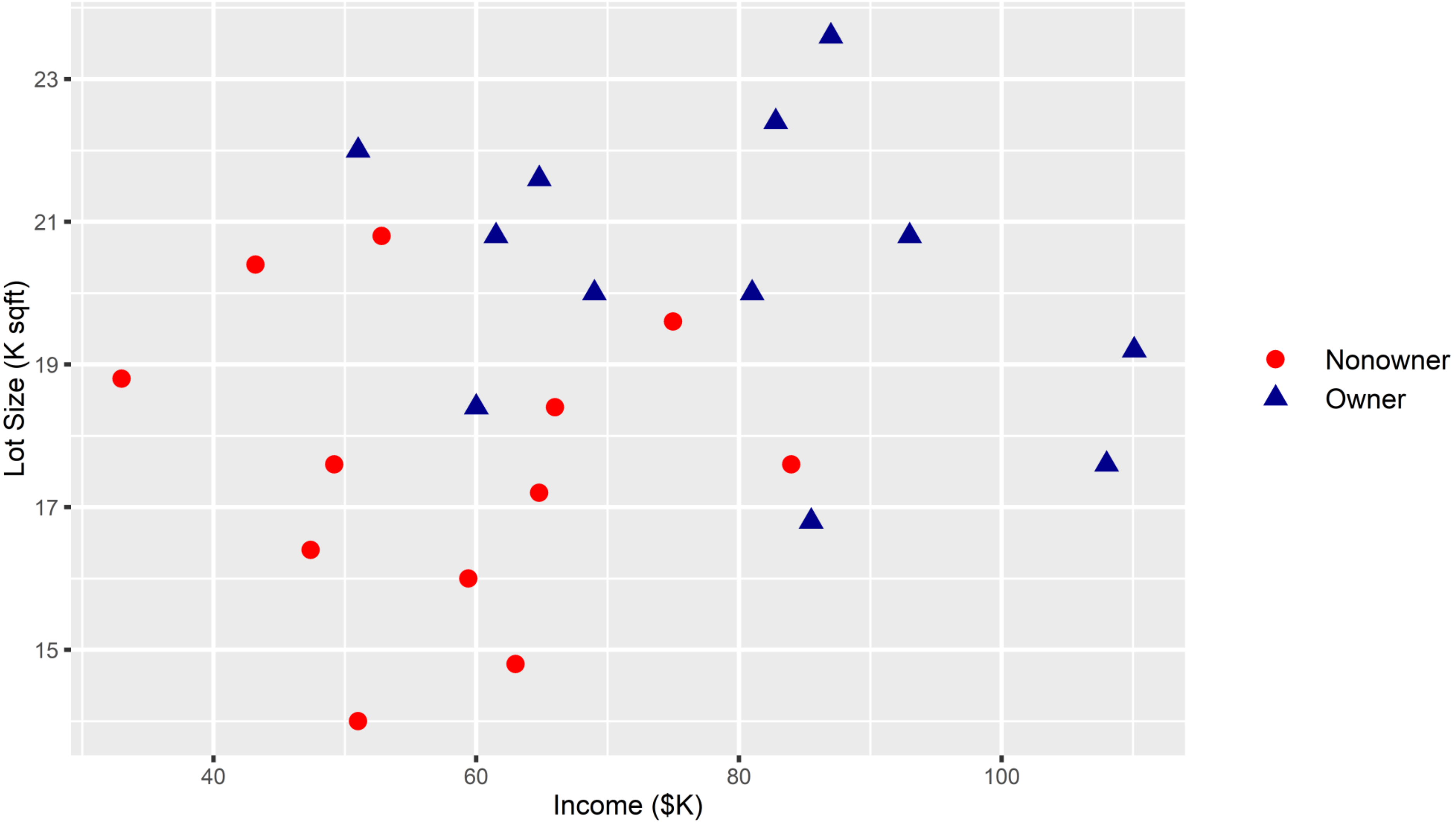
- Riding-mower manufacturer would like to find a way of classifying families in a city into an **owner** or **non-owner**
- Attributes
  - Income : Income of the household in thousand of dollars
  - Lot Size : Lot size in thousand of square foot
  - Ownership : Owner or Non-owner

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

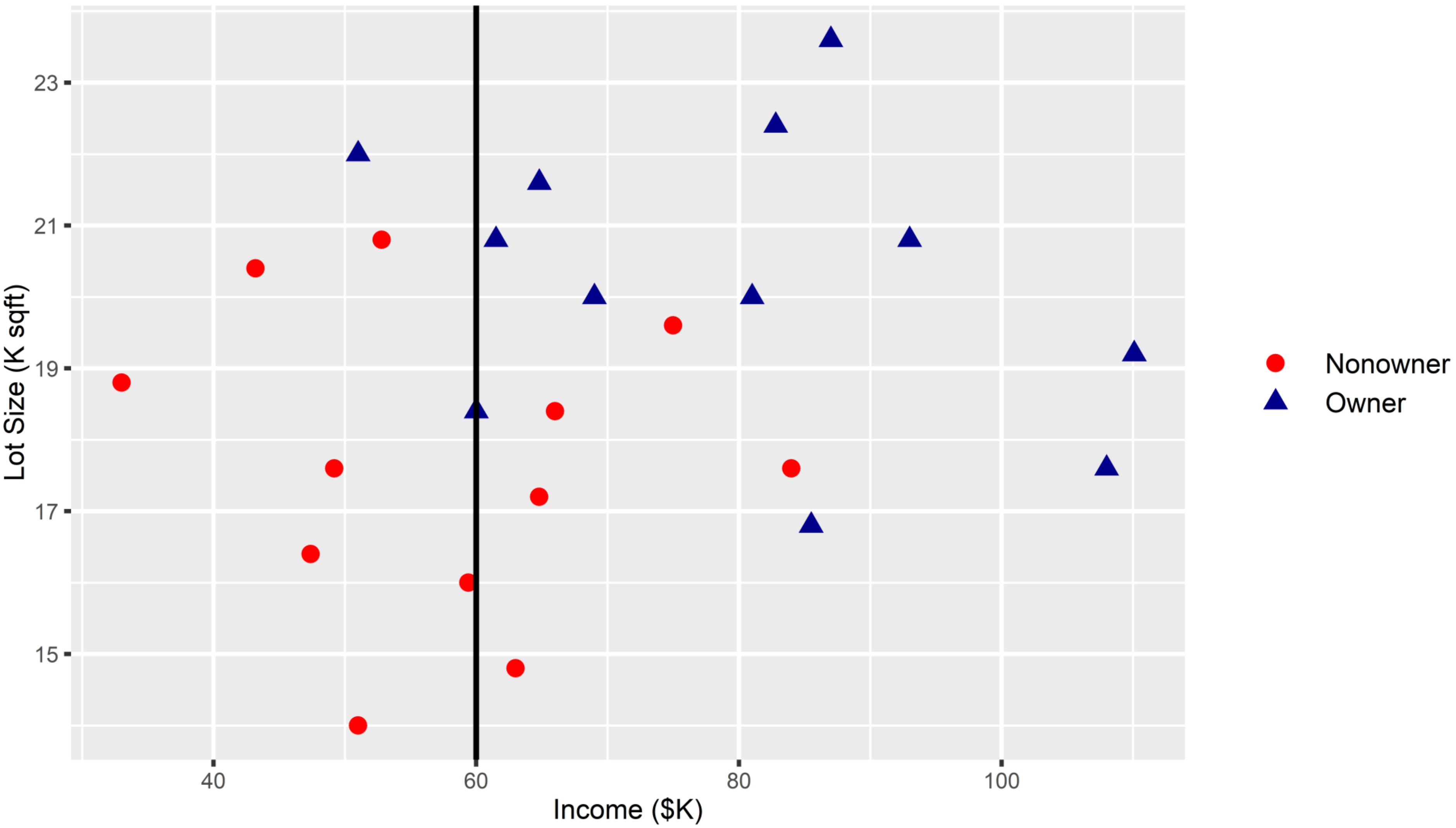
⋮

⋮

Scatter plot of entire data

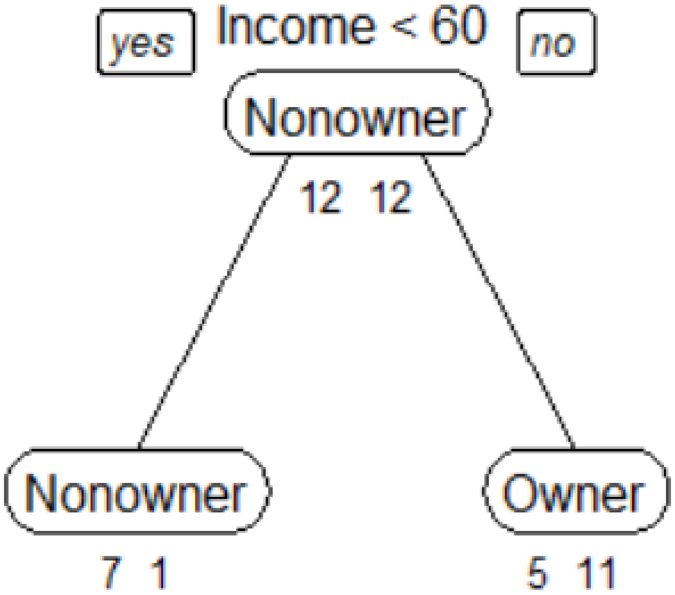


First split at Income = 60

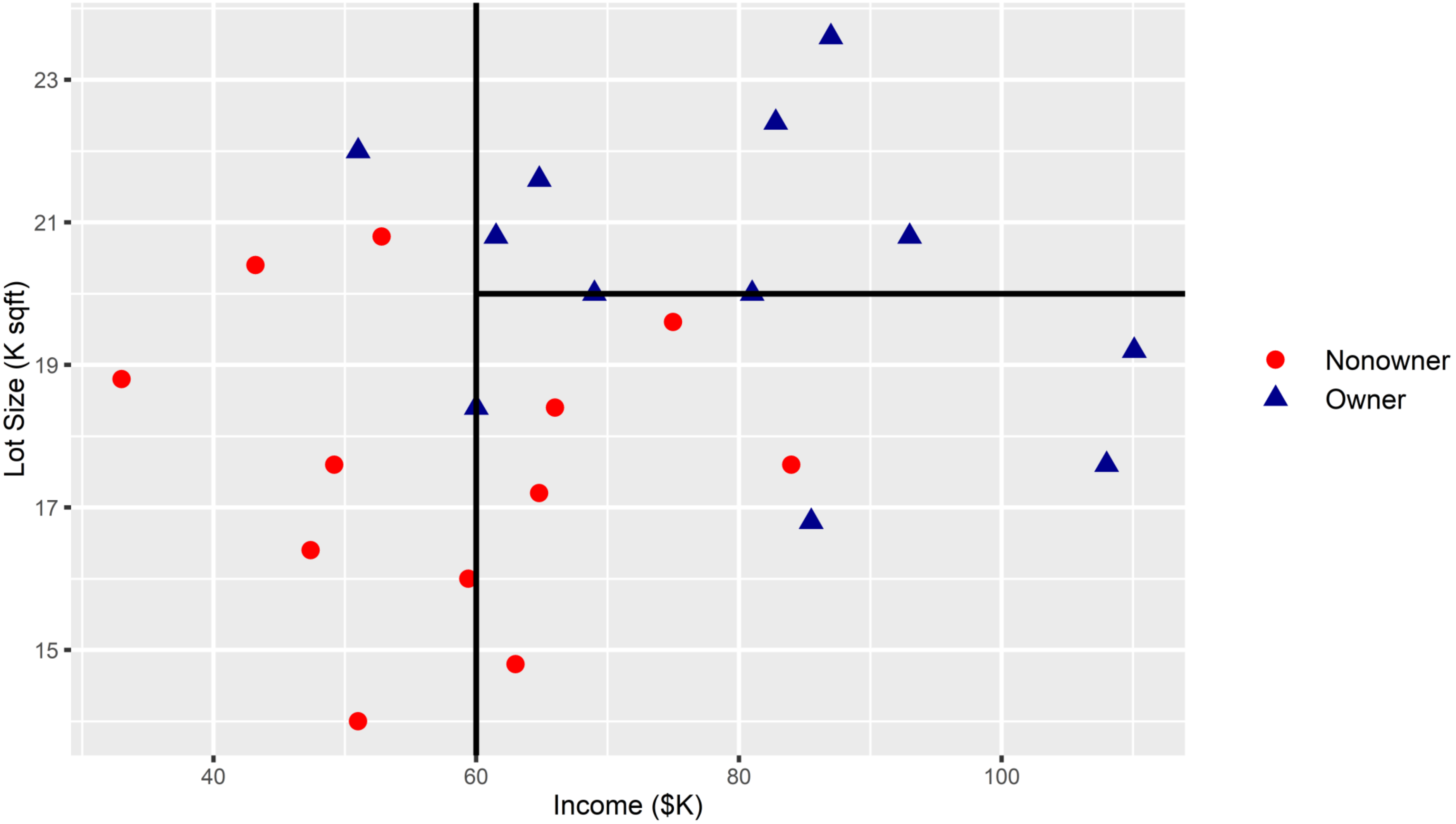


© Pradeep Pendem

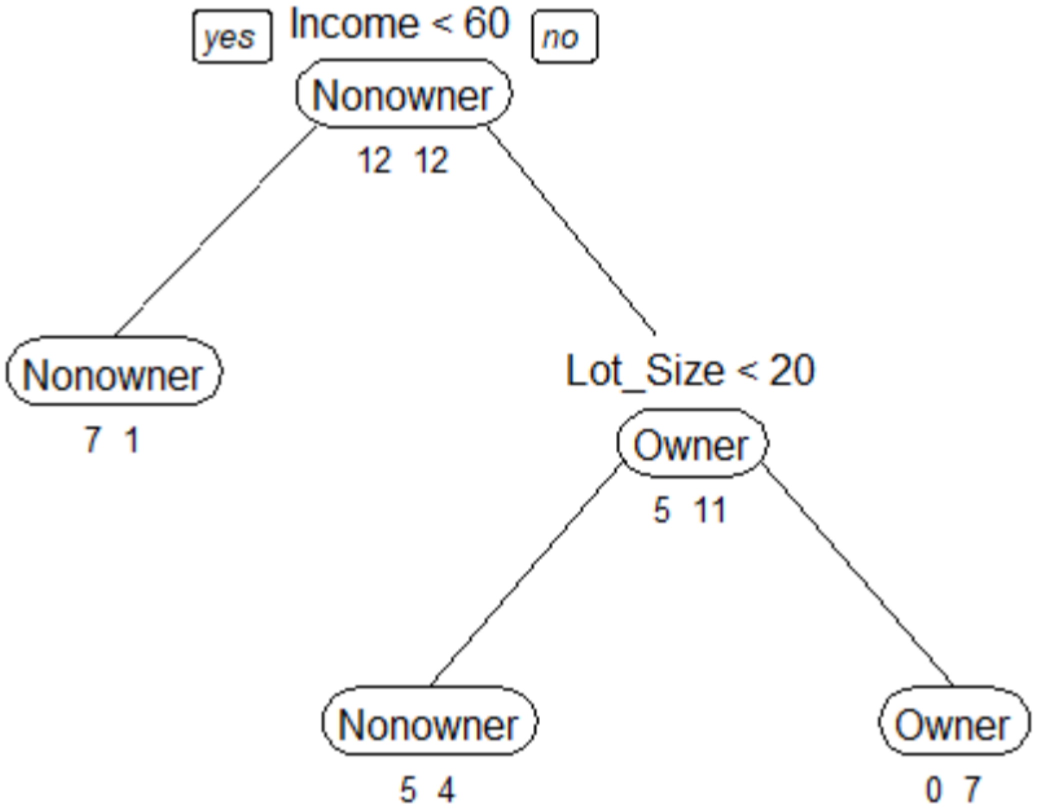
First split at Income = 60



Second split at Lot Size = 20

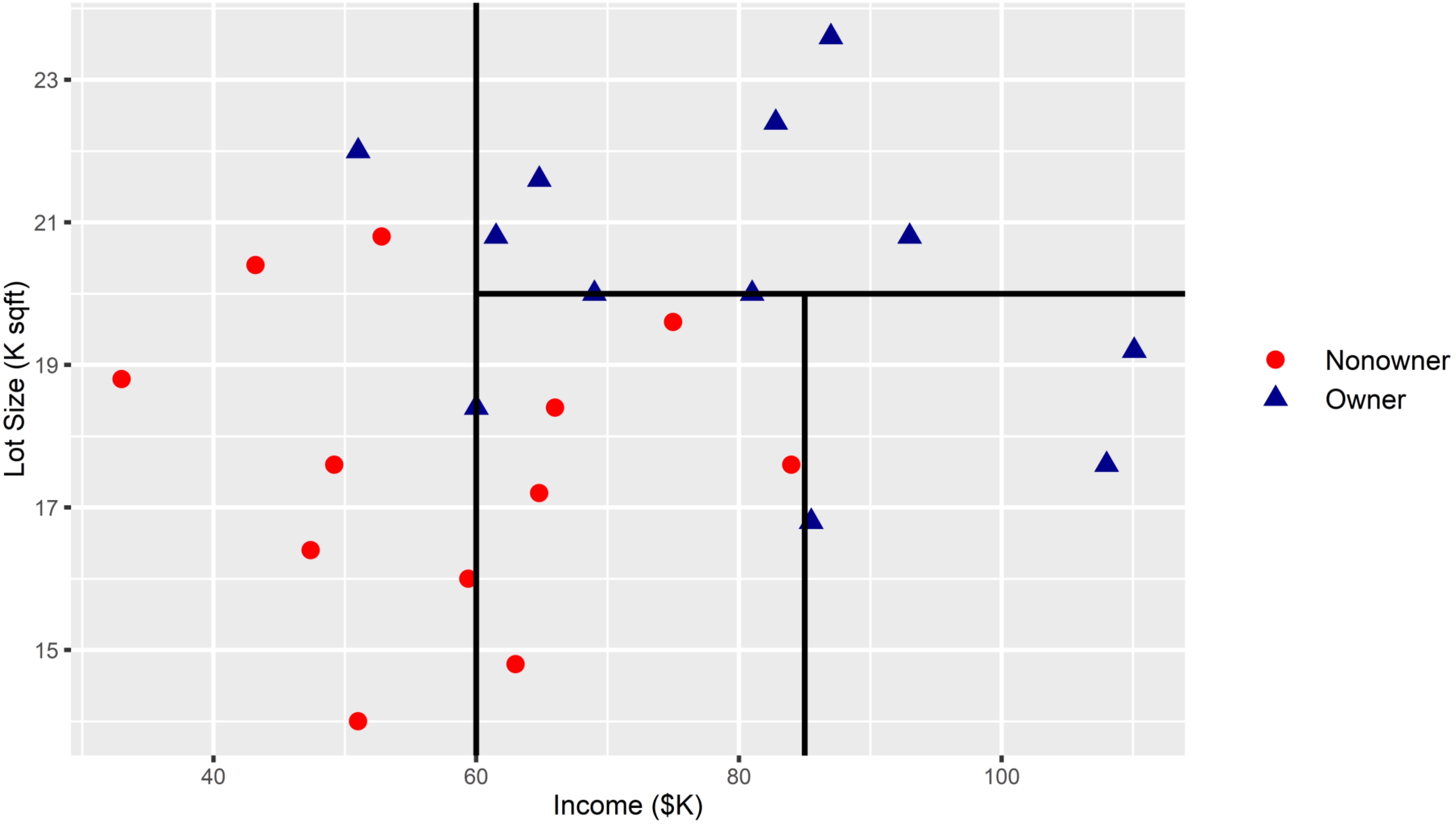


Second split at Lot Size = 20

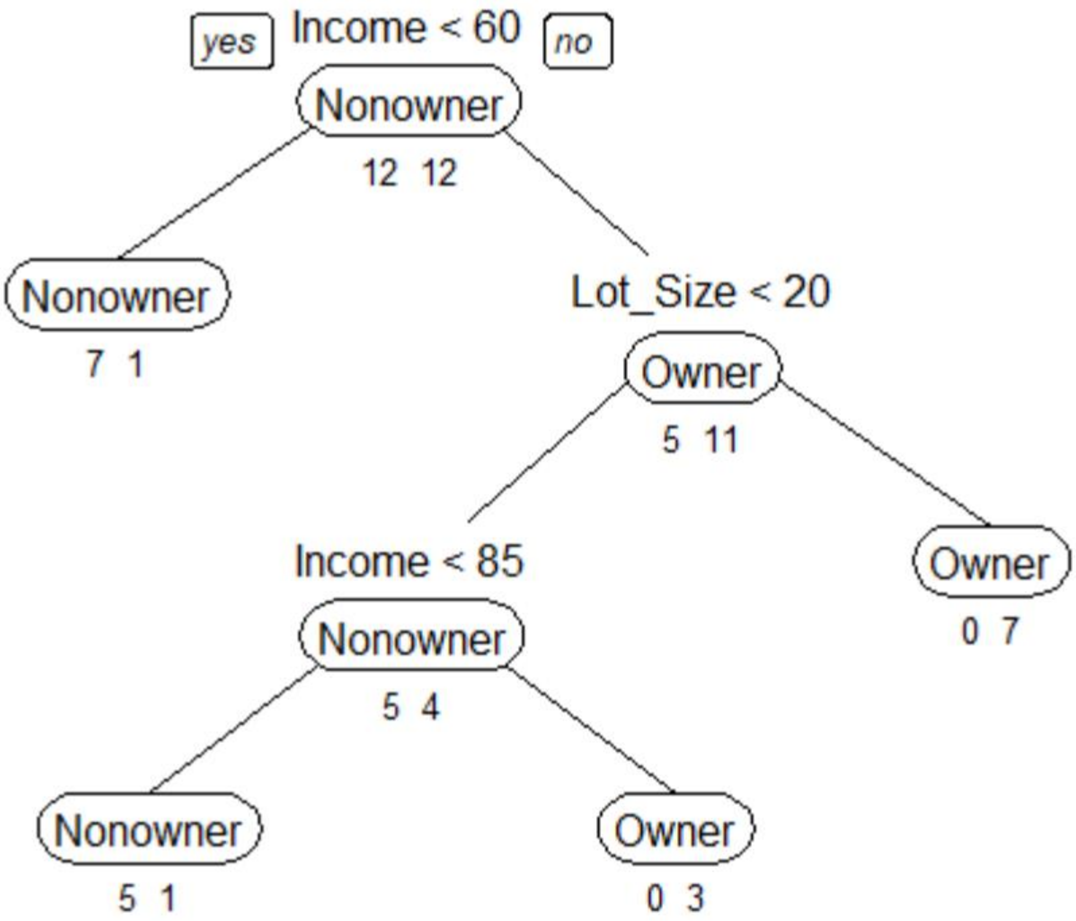


© Pradeep Pendem

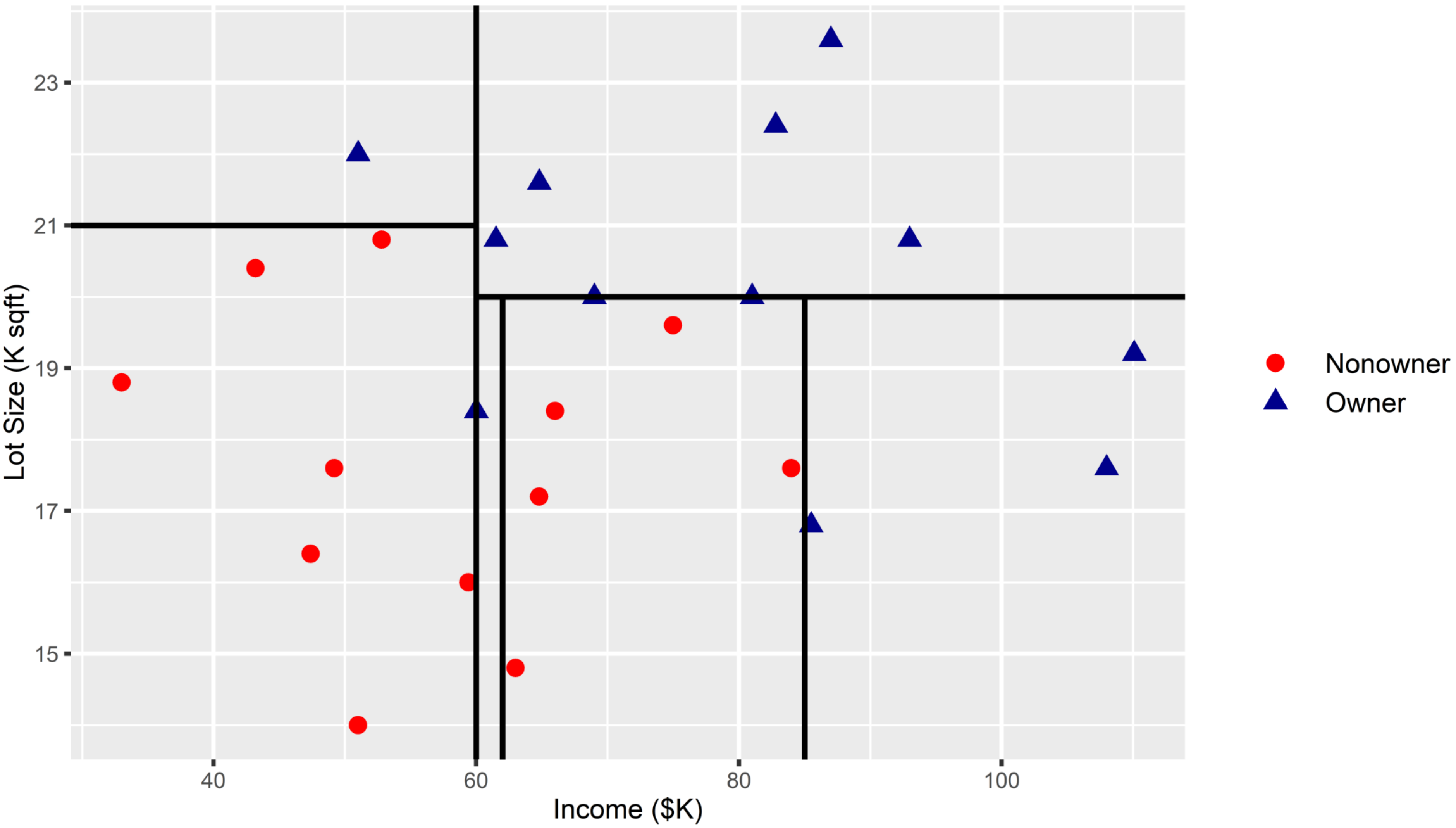
Third split at Income = 85



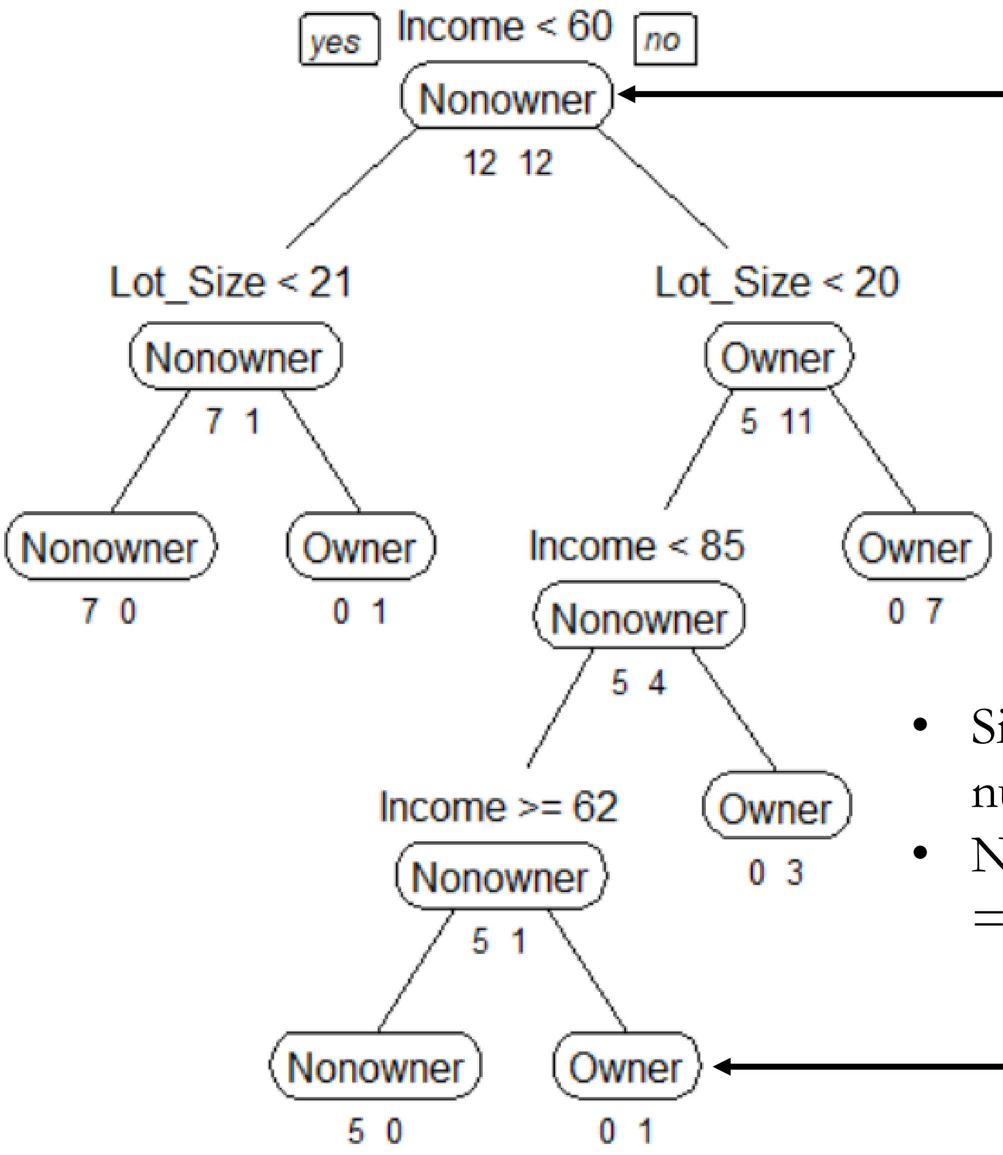
Third split at Income = 85



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

© Pradeep Pendem

Today’s class mandatory steps

- Create a folder name “**o. regression_tree**” within the folder

  “**oba_455_555_ddpm_r/rproject**”

- Download “**regression_tree_code.R**”, and all **CSV** files from canvas

- Place all downloaded files in

  “**oba_455_555_ddpm_r /rproject/o. regression_tree**”

- Open RStudio project

- Open “**regression_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
  - ……

Regression Trees

Data on used Toyota Corolla cars

- Output : Price
- Attributes
  - age_08_04 : age in months as of august 2004
  - km : accumulated kilometers on the odometer
  - fuel_type : fuel type (petrol, diesel, cng)
  - hp : horse power
  - Automatic, Doors, Quarterly tax

Pruning a tree

- Step 1: Set the seed, data partition - train & validation
- Step 2: Run a tree with options cp = 0.001, minsplit = 5 or 10, xval = 5 or 10
- Step 3: Plot the cp or relative error
- Step 4: Find the optimal cp where the error starts stabilizing
- Step 5: Prune the tree with the optimal cp
- Step 6: Predict the output variable in validation data
- Step 6: Generate accuracy measures

Pruning – Key options

- 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.

(Dis)Advantages of Trees

- Simple ; Requires little effort from users
- Useful for variable selection with most important predictors usually showing up at top of the tree
- Models non-linear and non-parametric
- Intrinsically robust to outliers
- Handle missing data without having to impute or delete records
- Sensitive to changes in the data – even a slight change can cause different splits
- Trees are relatively expensive to grow; Pruning adds a lot of time

Final Project (40%)

- Specify a business problem
- Identify a relevant dataset
- Business context could be in any area or function
- Assessment
  - Report (30%) + Presentation (10%)
- Presentation
  - 15-minute presentation on one of the classes of last week
  - **Presentation date(s) i**n the syllabus file

Final Report

- Formal report
    - Introduction, Problem description, Approach (Regression / Classification)
    - Data Analysis, Results, Inference
    - Conclusions, recommendations
- Regression: *k*-NN as Regression, Linear Regression & Regression Tree
- Classification: *k*-NN as classification, Logistic Regression & Classification Tree
- Assess the performance & recommend the best predictive model
- 8-10 pages including any tables and graphs (excluding code)
- Two or Three key insights from the entire analysis
- Submit the code with comments at end of the report

Public datasets for final project

kaggle

- https://www.kaggle.com/
- Online community of data scientists and machine learners
- Owned by Google Inc.
- Register yourself, and you can download datasets for free
- As of June 2017, Kaggle passed over 1,000,000 registered users
- Variety of datasets
- Your imagination only limits possibilities

Final Project presentation

- Presentation (10%)
  - 15-minute presentation followed by a 10-minute Q&A
  - **May 31st (Tue) & Jun 02nd (Thu)**
  - Groups are randomly assigned to the 2 days
  - Groups should send the ppt file by 8 am on their presentation date
  - Each member of the group should **mention the contribution** of their work in the last slide of the presentation file
- **Everyone** must be present in the class on the presentation days
  - Zero scores for presentation assessment if absent

May 31st presentations

- ACB
- ATJ
- HJJ
- P

Jun 02nd presentations

- AJA
- DJK
- MRV
- TAP

Next class

- Cluster Analysis

Thank You