4.2 - Decision Trees

Giovanni Della Lunga
giovanni.dellalunga@unibo.it

Introduction to Machine Learning for Finance

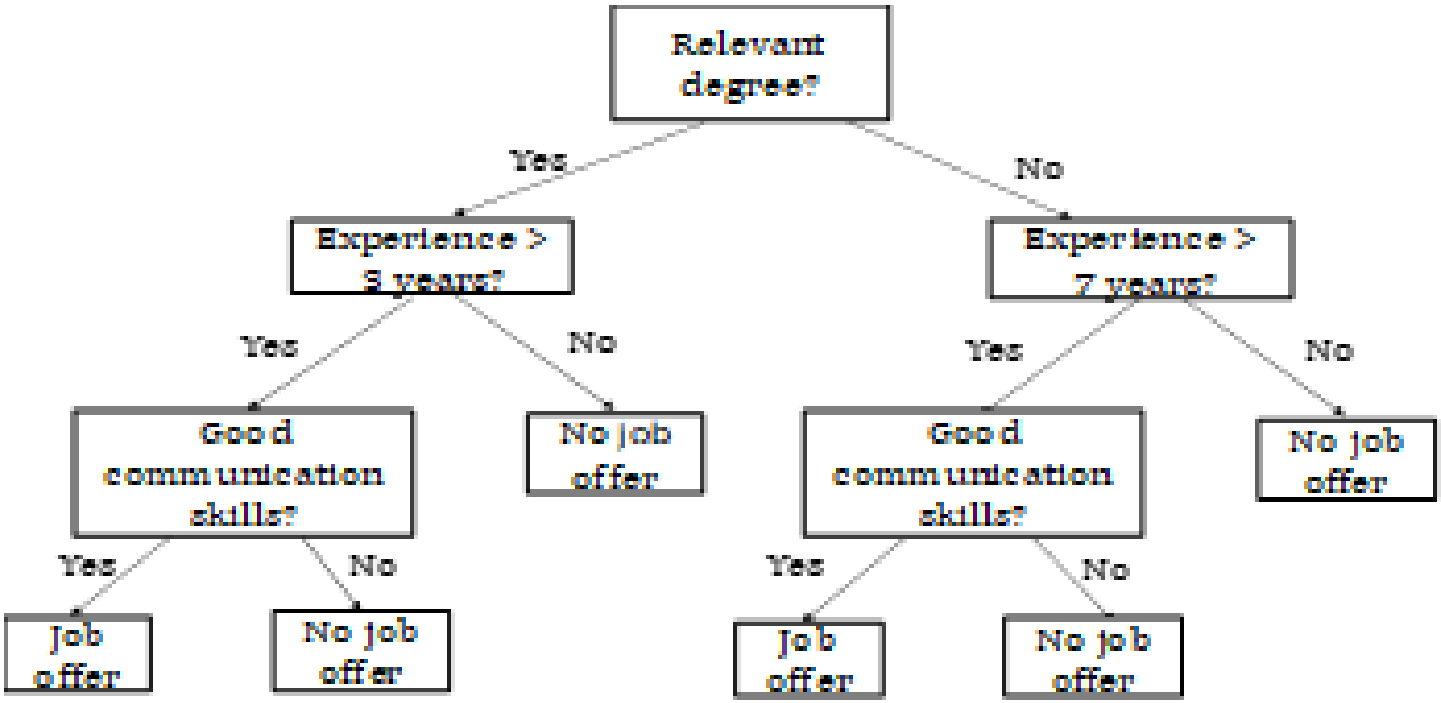
Bologna - February, 2022

Subsection 1

What is a decision tree

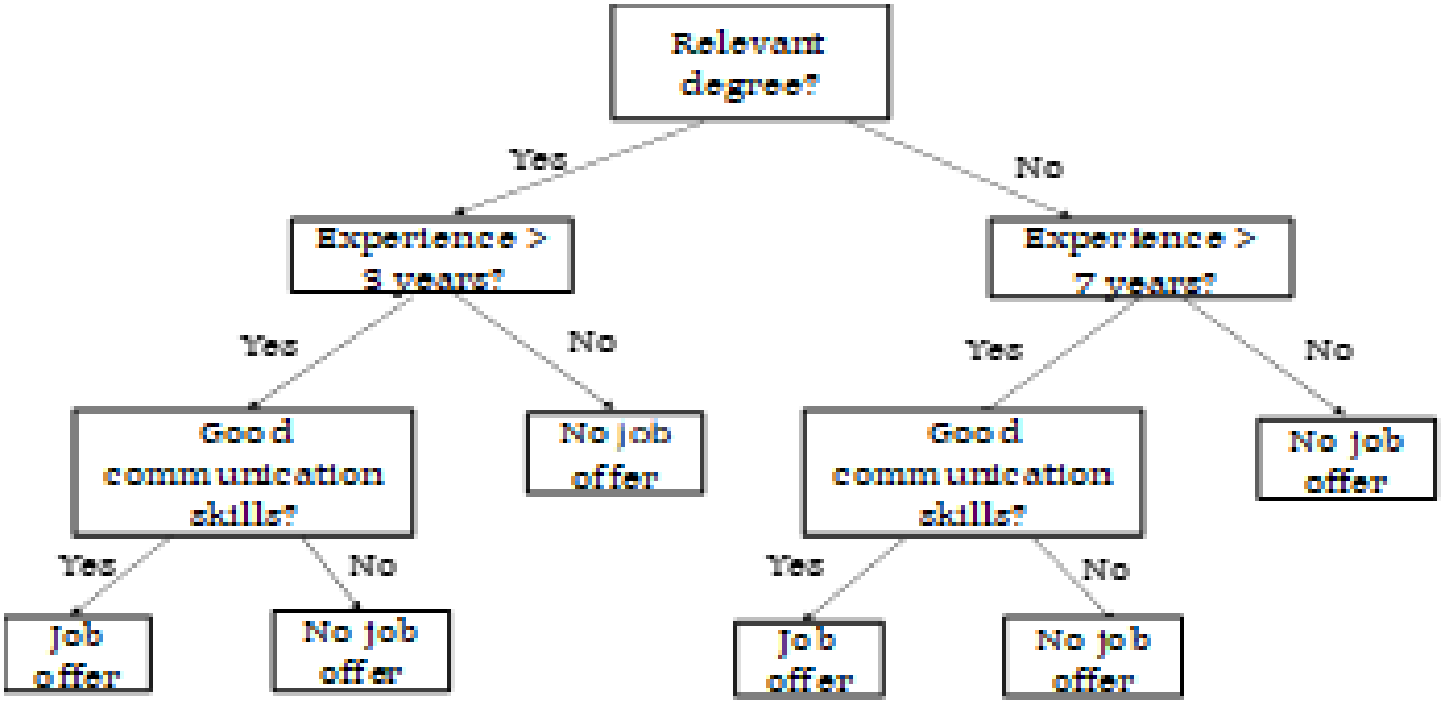
What is a Decision Tree

- A **Decision Tree** shows a step-by-step process for making predictions;
- Example of a Decision Tree to Determine Criterion for Hiring



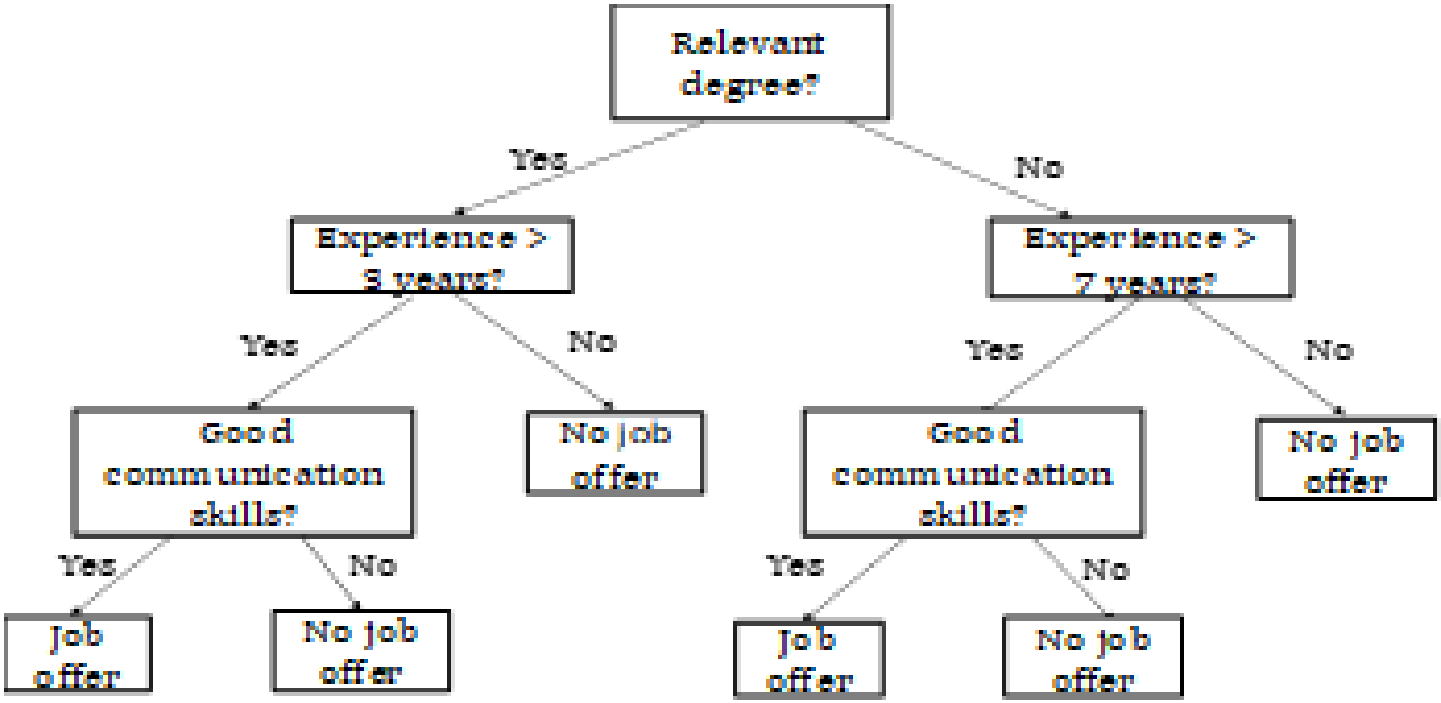
What is a Decision Tree

- The decision is made by **considering features one at a time rather than all at once**;
- The most important feature is considered first;



What is a Decision Tree

- What is the best feature to select at the root node?
- The feature to put at the root node is the one with the most **information gain**;



Measures of Uncertainty

- Suppose that there are n possible outcomes and p_i is the probability of outcome i with $\sum_{i=1}^{n} p_i = 1$
- Entropy measure of uncertainty:

$$\text{Entropy} = -\sum_{i=1}^{n} p_i \ln(p_i)$$

- Gini Measure of uncertainty:

$$\text{Gini} = 1 - \sum_{i=1}^{n} p_i^2$$

Information Gain

- The information gain is the expected decrease in uncertainty (as measured by either entropy or Gini).
- Suppose that there is a 20% chance that a person will receive a job offer
- Suppose further that there is a 50% chance the person has a relevant degree. If the person does have a relevant degree the probability of a job offer rises to 30%, otherwise it falls to 10%

Information Gain

$$\text{Initial Entropy} = -[0.2\ \log(0.2) + 0.8\ \log(0.8)] = 0.7219$$

$$\begin{aligned}\text{Expected Entropy} = &- 0.5 \cdot [0.1 \log(0.1) + 0.9 \log(0.9)] \\ &- 0.5 \cdot [0.3 \log(0.3) + 0.7 \log(0.7)] \\ &= 0.6751\end{aligned}$$

The Expected information gain from knowing whether there is a relevant degree is

$$\text{gain} = 0.7219 - 0.6751 = 0.0468$$

The Decision Tree Algorithm

- Algorithm **chooses the feature at the root of the tree that has the greatest expected information gain**;
- At subsequent nodes it choose the feature (not already chosen) that has the greatest expected information gain;
- When there is a threshold, it determines the optimal threshold for each feature (i.e., the threshold that maximizes the expected information gain for that feature) and bases calculations on that threshold

Subsection 2

Application to credit decision

Lending Club Example

- Choosing the root node when there are four features:

$$\text{Initial Entropy} = -0.8276 \times \log(0.8276) - 0.1724 \times \log(0.1724) = 0.6632$$

Feature	Threshold value	Expected entropy	Expected Information gain
Home Ownership	N.A.	0.6611	0.0020
Income	$85,202	0.6573	0.0058
Debt to income ratio	19.87	0.6601	0.0030
FICO credit score	717.5	0.6543	0.0088

Lending Club Example

Choosing the next node if $FICO > 717.5$

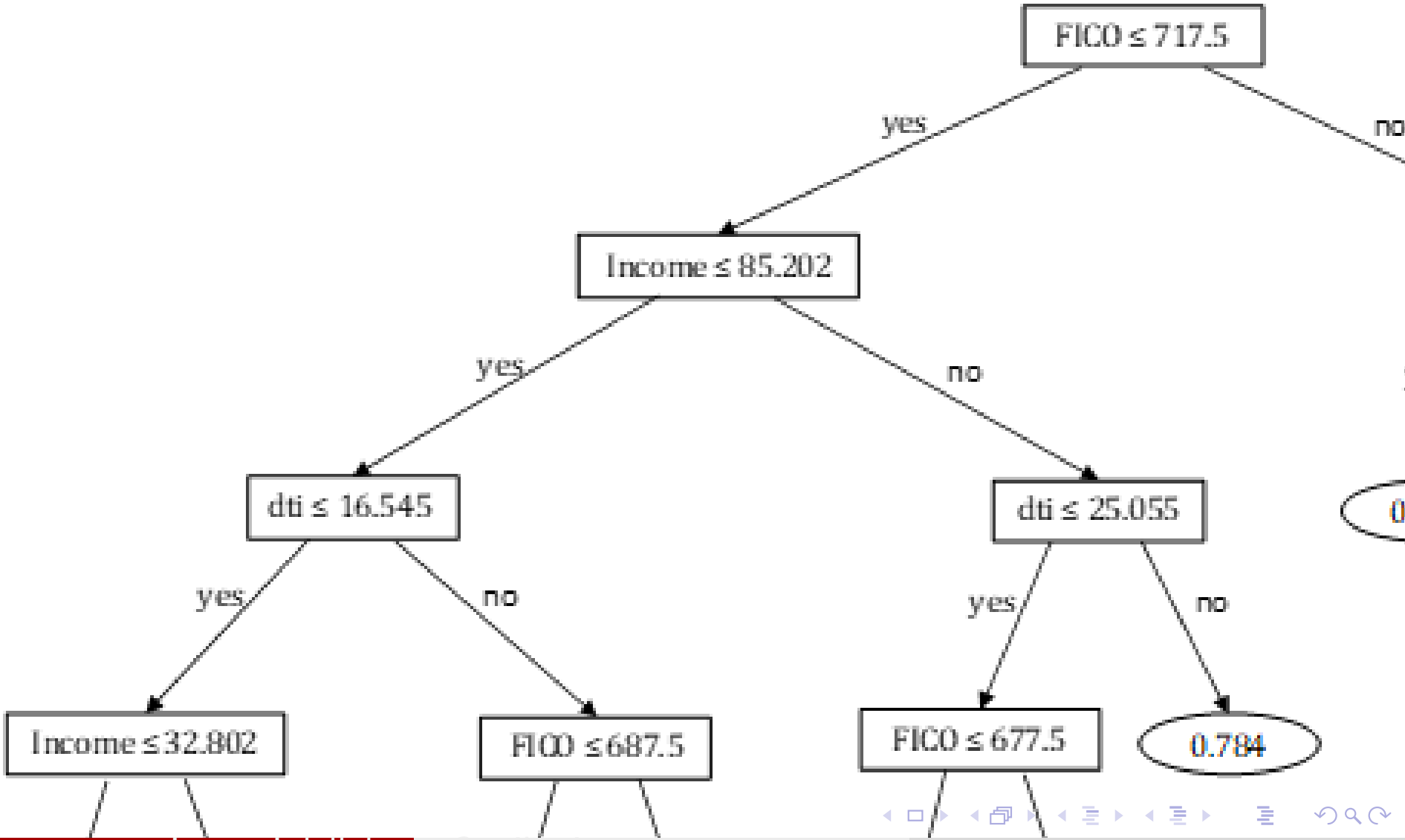
Feature	Threshold value	Expected entropy	Information gain
Home ownership	N.A.	0.4400	0.0003
Income ($'000s)	48.75	0.4330	0.0072
Debt to income (%)	21.13	0.4379	0.0023
FICO score	789	0.4354	0.0048

Lending Club Example

Choosing the next node if $FICO \leq 717.5$

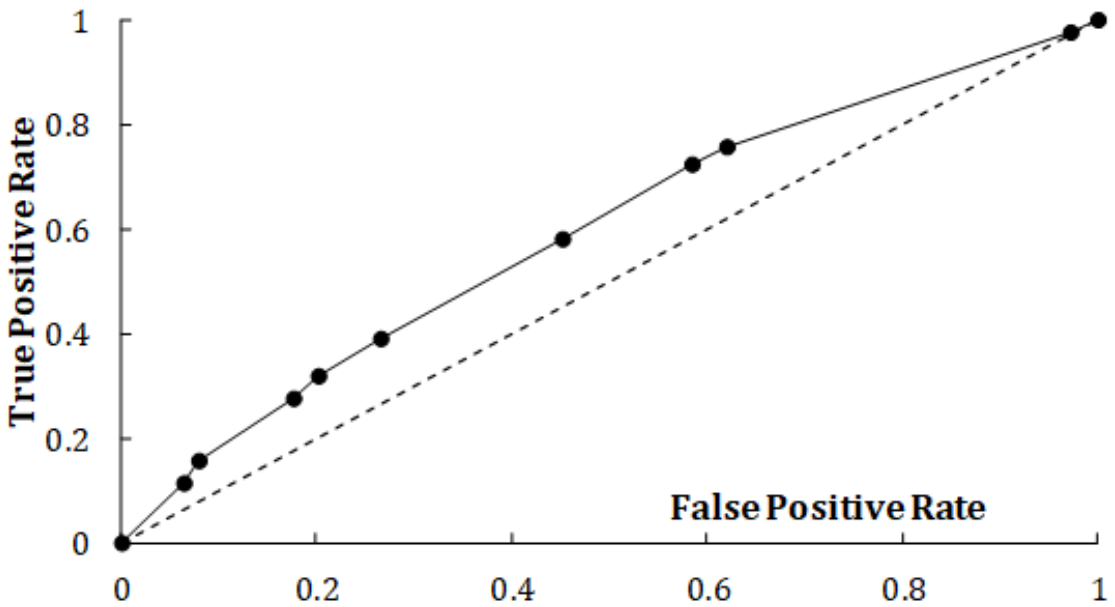
Feature	Threshold value	Expected entropy	Information gain
Home ownership	N.A.	0.7026	0.0017
Income ($'000s)	85.202	0.6989	0.0055
Debt to income (%)	16.80	0.7013	0.0030
FICO score	682	0.7019	0.0025

Lending Club Example: The Tree



Choosing a criterion for accepting loans

As with logistic regression we can choose to accept loans where the probability of a good loan is above some threshold, Z



Naive Bayes Classifier

From Bayes theorem:

$$\mathrm{P}(C|x_1, x_2, \ldots, x_n) = \frac{\mathrm{P}(x_1, x_2, \ldots, x_n|C)}{\mathrm{P}(x_1, x_2, \ldots, x_n)}\mathrm{P}(C) \tag{1}$$

If the features x_i are (approximately) independent this reduces to:

$$\mathrm{P}(C|x_1, x_2, \ldots, x_n) = \frac{\mathrm{P}(x_1|C)\mathrm{P}(x_2|C)\cdots\mathrm{P}(x_n|C)}{\mathrm{P}(x_1, x_2, \ldots, x_n)}\mathrm{P}(C) \tag{2}$$

Example

The unconditional probability of a good loan is 0.85.There are 3 independent features:

- **Whether the applicant owns a house (denoted by H)**. The probability of the applicant owning her own house if the loan is good is 60% where the probability of the applicant owning her own house if the loan defaults is 50%.
- **Whether the applicant has been employed for more than one year (denoted by E)**. The probability of the applicant being employed for more than one year if the loan is good is 70% whereas the probability of this if the loan defaults is 60%.
- **Whether there are two applicants rather than only one (denoted by T)**. The probability of two applicants when the loan is good is 20% whereas the probability of two applicants when the loan defaults is 10%.

Example (cont.)

- $\text{Prob}(\text{Good Loan}|\text{H, E, T}) = \frac{0.6 \times 0.7 \times 0.2}{\text{Prob}(\text{H and E and T})} \times 0.85 = \frac{0.0714}{\text{Prob}(\text{H and E and T})}$
- $\text{Prob}(\text{Defaulting Loan}|\text{H, E, T}) = \frac{0.5 \times 0.6 \times 0.1}{\text{Prob}(\text{H and E and T})} \times 0.15 = \frac{0.0045}{\text{Prob}(\text{H and E and T})}$
- But these probabilities must sum to one so conditional on H, E, and T, the probability of a good loan is

$$\frac{0.0714}{0.0714 + 0.0045} = 0.941$$

Naive Bayes classifier and Continuous Features

Assume normal distributions and consider someone who has a FICO score of 720 and an income of 100

Loan result	Mean FICO	SD FICO	Mean Income	SD Income
Good loan	696.19	31.29	79.83	59.24
Defaulting loan	686.65	24.18	68.47	48.81

Calculations

- Probability density for FICO conditional on good loan:

$$\frac{1}{\sqrt{2\pi} \times 31.29} \exp\left(-\frac{(720 - 696.19)^2}{2 \times 31.29^2}\right) = 0.00954$$

- Probability density for income conditional on good loan:

$$\frac{1}{\sqrt{2\pi} \times 59.24} \exp\left(-\frac{(100 - 79.83)^2}{2 \times 59.24^2}\right) = 0.00636$$

- Probability density for FICO conditional default:

$$\frac{1}{\sqrt{2\pi} \times 24.18} \exp\left(-\frac{(720 - 686.65)^2}{2 \times 24.18^2}\right) = 0.00637$$

- Probability density for income conditional on default:

$$\frac{1}{\sqrt{2\pi} \times 48.81} \exp\left(-\frac{(100 - 68.47)^2}{2 \times 48.81^2}\right) = 0.00663$$

Calculations continued

- The unconditional probability of a good loan is 0.8276
- The probability of a good loan conditional on FICO and income is

$$\frac{0.00954 \times 0.00636 \times 0.8276}{Q} = \frac{5.020 \times 10^{-5}}{Q}$$

  where Q is the probability density of the observed FICO and income.
- The probability of a defaulting loan conditional on FICO and income is

$$\frac{0.00637 \times 0.00663 \times 0.1724}{Q} = \frac{0.729 \times 10^{-5}}{Q}$$

- Probability of a good loan is $5.020/(5.020 + 0.729) = 0.873$

Subsection 3

Continuous target variables

Continuous Target Variables

- We can construct a tree where instead of maximizing expected information gain we maximize the expected decrease in mean squared error
- Consider the Iowa house price (000s) example where we only consider Overall Quality and Living Area
- For root node we obtain

Feature	Threshold, Q	No. Obs ≤Q	mse of Obs ≤Q	No. obs > Q	mse obs >Q	Exp mse
Overall Quality	7.5	1,512	2,376	288	7,312	3,166
Living Area	1,482	949	1,451	851	6,824	3,991

Continuous Target Variables

Second Level split when Overall Quality ≤ 7.5

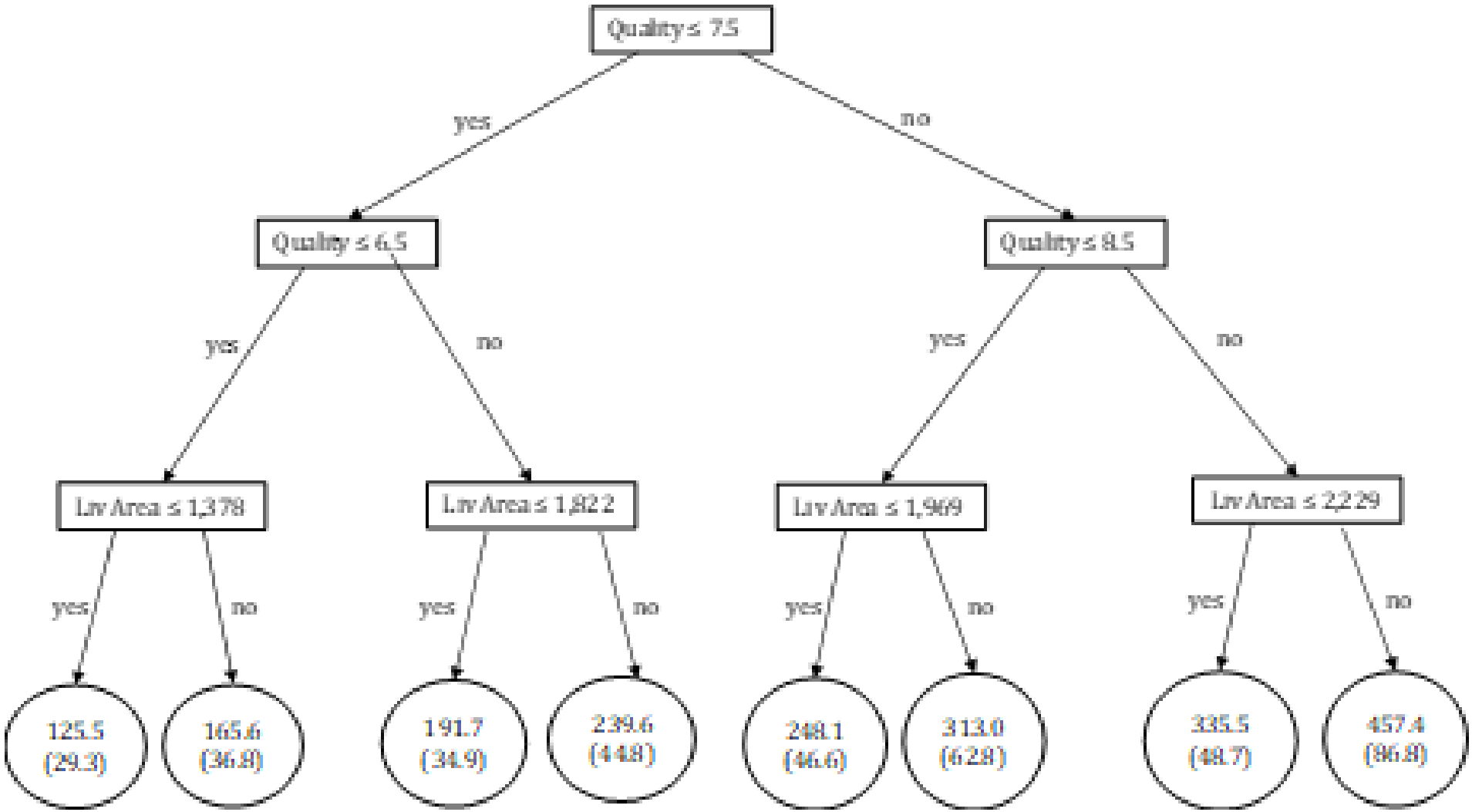
Feature	*Q*	*No. of obs* $\leq Q$	*mse of obs* $\leq Q$	*No. of obs* $> Q$	*mse of obs*$> Q$	*E(mse)*
Overall Quality	6.5	1,122	1,433	390	1,939	1,564
Living (sq. ft.)	1,412	814	1,109	698	2,198	1,612

Continuous Target Variables

Second Level split when Overall Quality > 7.5

Feature	*Q*	*No. of obs ≤ Q*	*mse of obs ≤ Q*	*No. of obs > Q*	*mse of obs> Q*	*E(mse)*
Overall Quality	8	214	3,857	74	8,043	4,933
Living (sq. ft.)	1,971	165	3,012	123	8,426	5,324

Continuous Target Variables: The Tree



Subsection 4

Ensemble methods

Random Forest

- This involves constructing many trees by for example:
- Using samples bootstrapped from the original data
- Using a random subset of features at each node
- Randomizing thresholds in some way
- The final decision can be a majority vote or a weighted majority vote. Weights can reflect probability estimates (when available) or evidence from a hold-out test data set.

Ensemble

- More generally the results from several different ML algorithms can be combined to obtain a single estimate.
- Many weak learners can sometimes be combined into a strong learner
- The extent to which this is possible depends on correlations between learners

Bagging

- Sample with replacement to create new data sets
- Use voting or averaging methods for final estimate

Boosting

- Predictions are made sequentially, each trying to correct the previous error
- One approach (AdaBoost) increases the weight given to misclassified observations
- Another approach (Gradient boosting) tries to fit a new predictor to the error made by the previous predictor