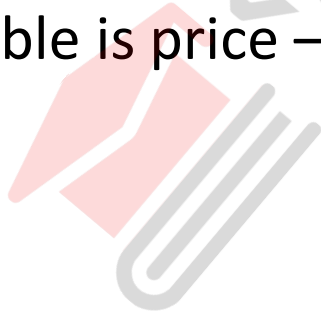


Regression model

- Relation between variables where changes in some variables may “explain” or possibly “cause” changes in other variables.
- Explanatory variables are termed the **independent** variables and the variables to be explained are termed the **dependent** variables.
- Regression model estimates the nature of the relationship between the independent and dependent variables.
 - Change in dependent variables that results from changes in independent variables, ie. size of the relationship.
 - Strength of the relationship.
 - Statistical significance of the relationship.

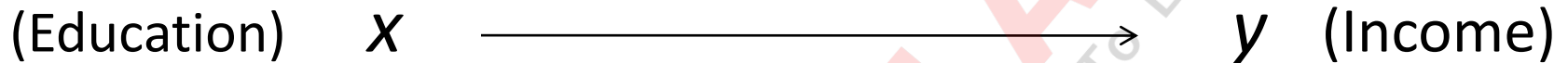
Examples

- Dependent variable is employment income – independent variables might be hours of work, education, occupation, gender, age, region, years of experience, unionization status, etc.
- Price of a product and quantity produced or sold:
 - Quantity sold affected by price. Dependent variable is quantity of product sold – independent variable is price.
 - Price affected by quantity offered for sale. Dependent variable is price – independent variable is quantity sold.

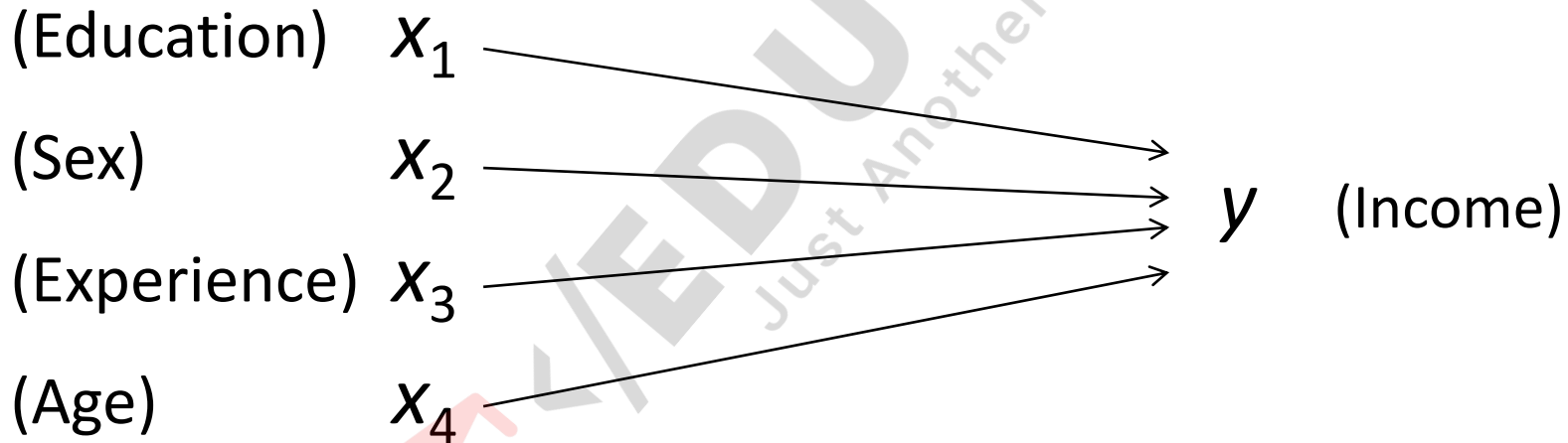


Bivariate and multivariate models

Bivariate or simple regression model



Multivariate or multiple regression model



Model with simultaneous relationship



Bivariate or simple linear regression (ASW, 466)

- x is the independent variable
- y is the dependent variable
- The regression model is

$$y = \beta_0 + \beta_1 x + \varepsilon$$

- The model has two variables, the independent or explanatory variable, x , and the dependent variable y , the variable whose variation is to be explained.
- The relationship between x and y is a linear or straight line relationship.
- Two parameters to estimate – the slope of the line β_1 and the y -intercept β_0 (where the line crosses the vertical axis).
- ε is the unexplained, random, or error component

Regression line

- The regression model is $y = \beta_0 + \beta_1 x + \varepsilon$
- Data about x and y are obtained from a sample.
- From the sample of values of x and y , estimates b_0 of β_0 and b_1 of β_1 are obtained using the least squares or another method.
- The resulting estimate of the model is

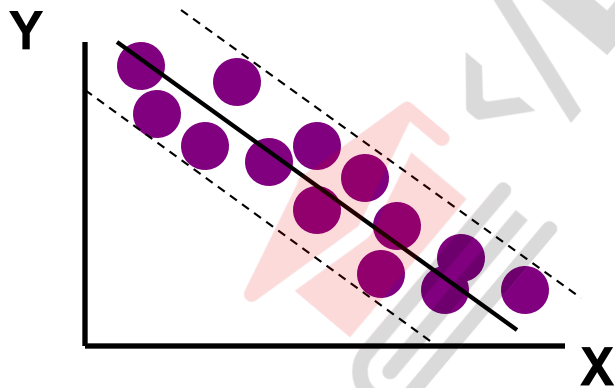
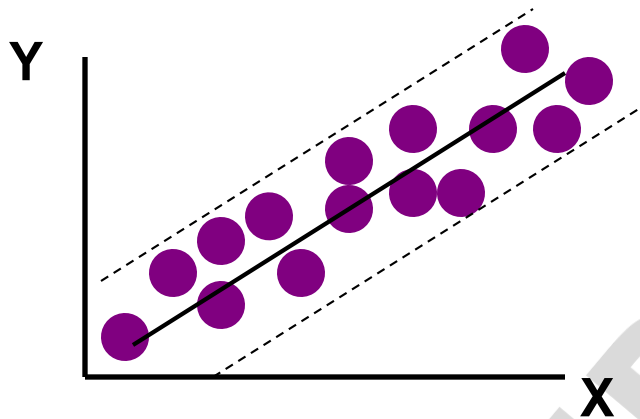
$$\hat{y} = b_0 + b_1 x$$

- The symbol \hat{y} is termed “y hat” and refers to the predicted values of the dependent variable y that are associated with values of x , given the linear model.

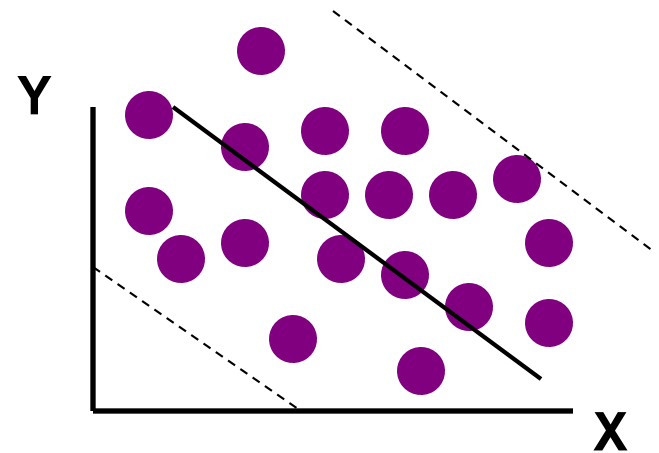
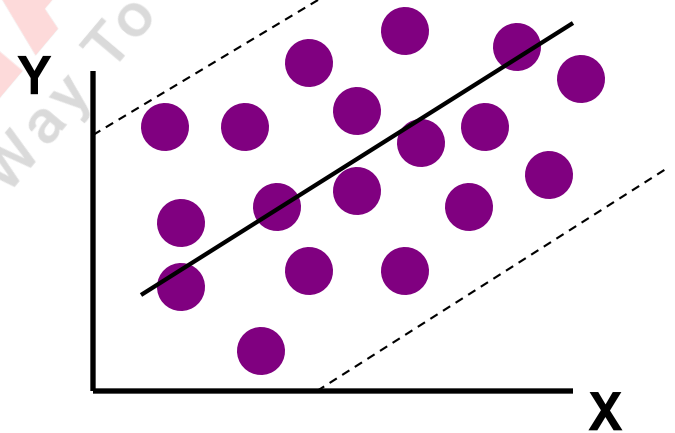
Types of Relationships

(continued)

Strong relationships



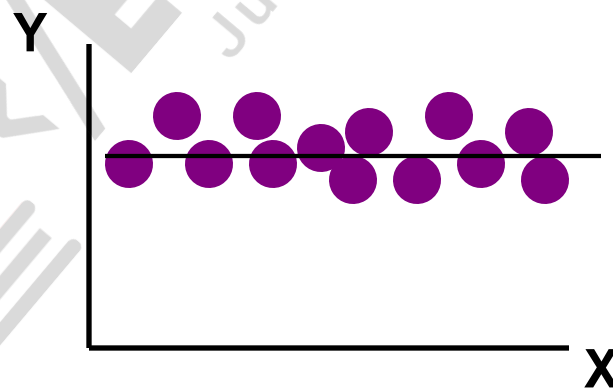
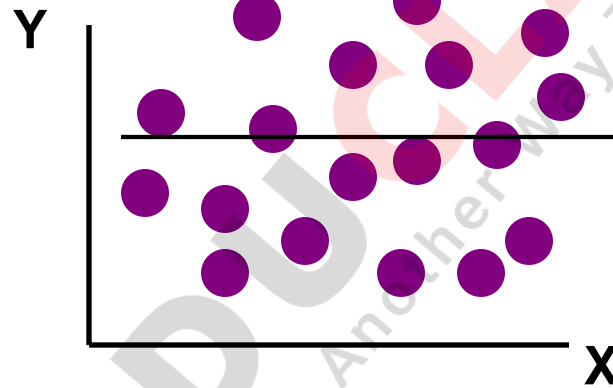
Weak relationships



Types of Relationships

(continued)

No relationship



Uses of regression

- Amount of change in a dependent variable that results from changes in the independent variable(s) – can be used to estimate elasticities, returns on investment in human capital, etc.
- Attempt to determine causes of phenomena.
- Prediction and forecasting of sales, economic growth, etc.
- Support or negate theoretical model.
- Modify and improve theoretical models and explanations of phenomena.

Simple Regression Model

- Make prediction about the starting salary of a current college graduate
- Data set of starting salaries of recent college graduates

Data Set

Annual starting salary (dollars)

20,000

24,500

23,000

25,000

20,000

22,500

Total 135,000

Compute Average Salary

$$\bar{Y} = \frac{\sum_{i=1}^n Y_i}{n} = \frac{\$135,000}{6} = \$22,500$$

**How certain are of this prediction?
There is variability in the data.**

Simple Regression Model

- Use total variation as an index of uncertainty about our prediction

Compute Total Variation

Y_i	$(Y_i - \bar{Y})$	$(Y_i - \bar{Y})^2$
20,000	-2,500	6,250,000
24,500	2,000	4,000,000
23,000	500	250,000
25,000	2,500	6,250,000
20,000	-2,500	6,250,000
22,500	0	0
		<hr/>
		23,000,000

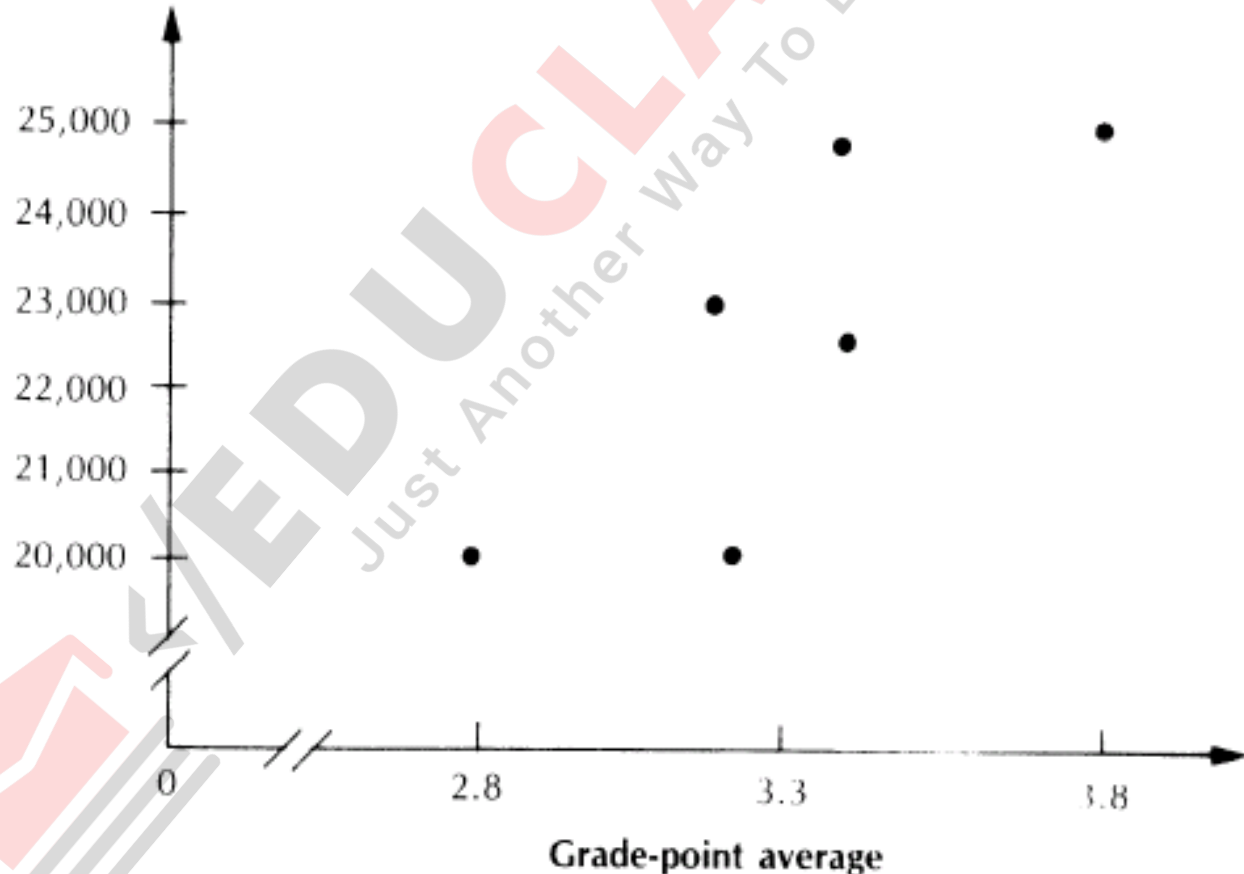
Total amount of variation $\sum_{i=1}^n (Y_i - \bar{Y})^2 = 23,000,000.$

- The smaller the amount of total variation the more accurate (certain) will be our prediction.

Simple Regression Model

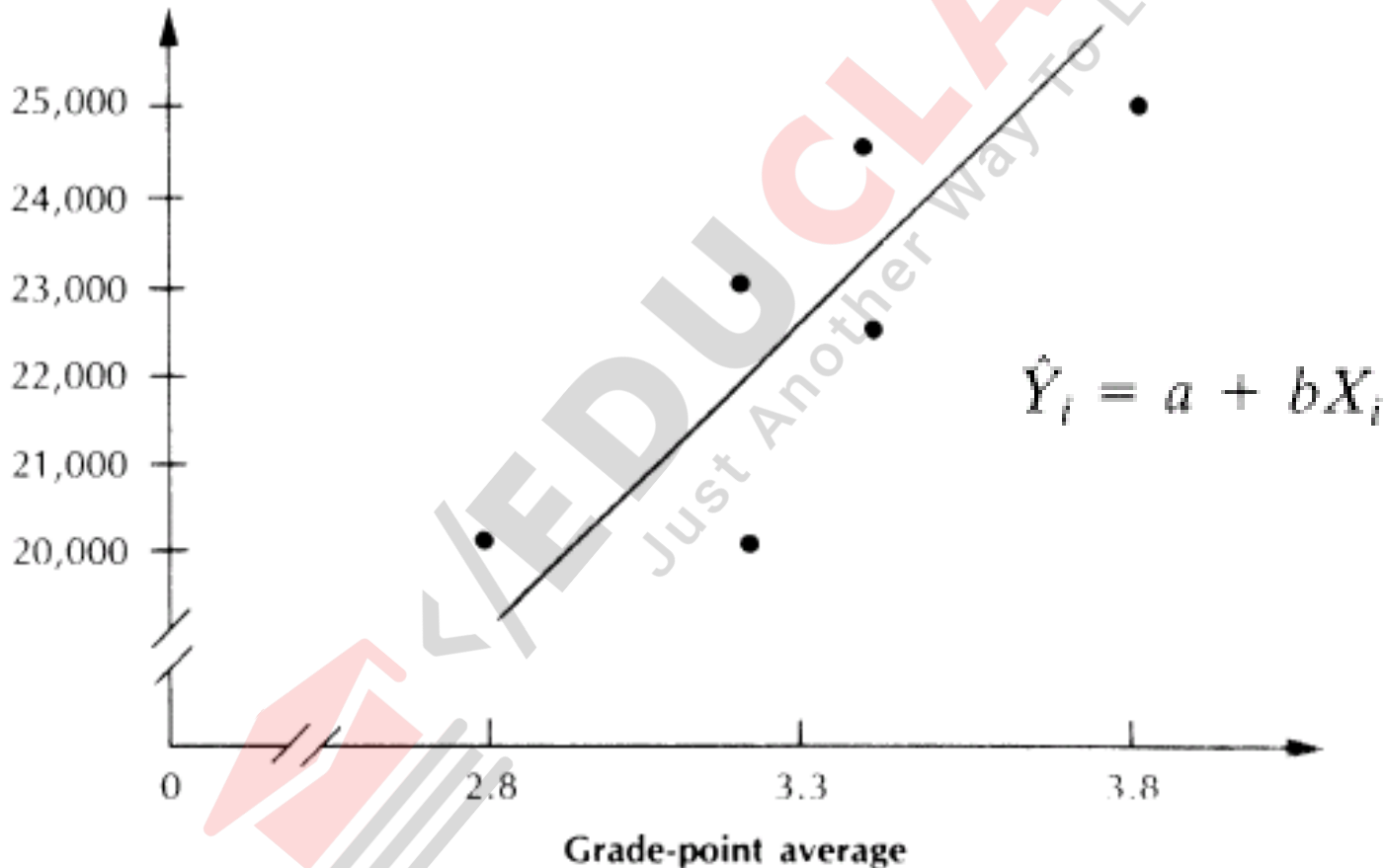
- How “explain” the variability - Perhaps it depends on the student’s GPA

Salary	GPA
20,000	2.8
24,500	3.4
23,000	3.2
25,000	3.8
20,000	3.2
22,500	3.4



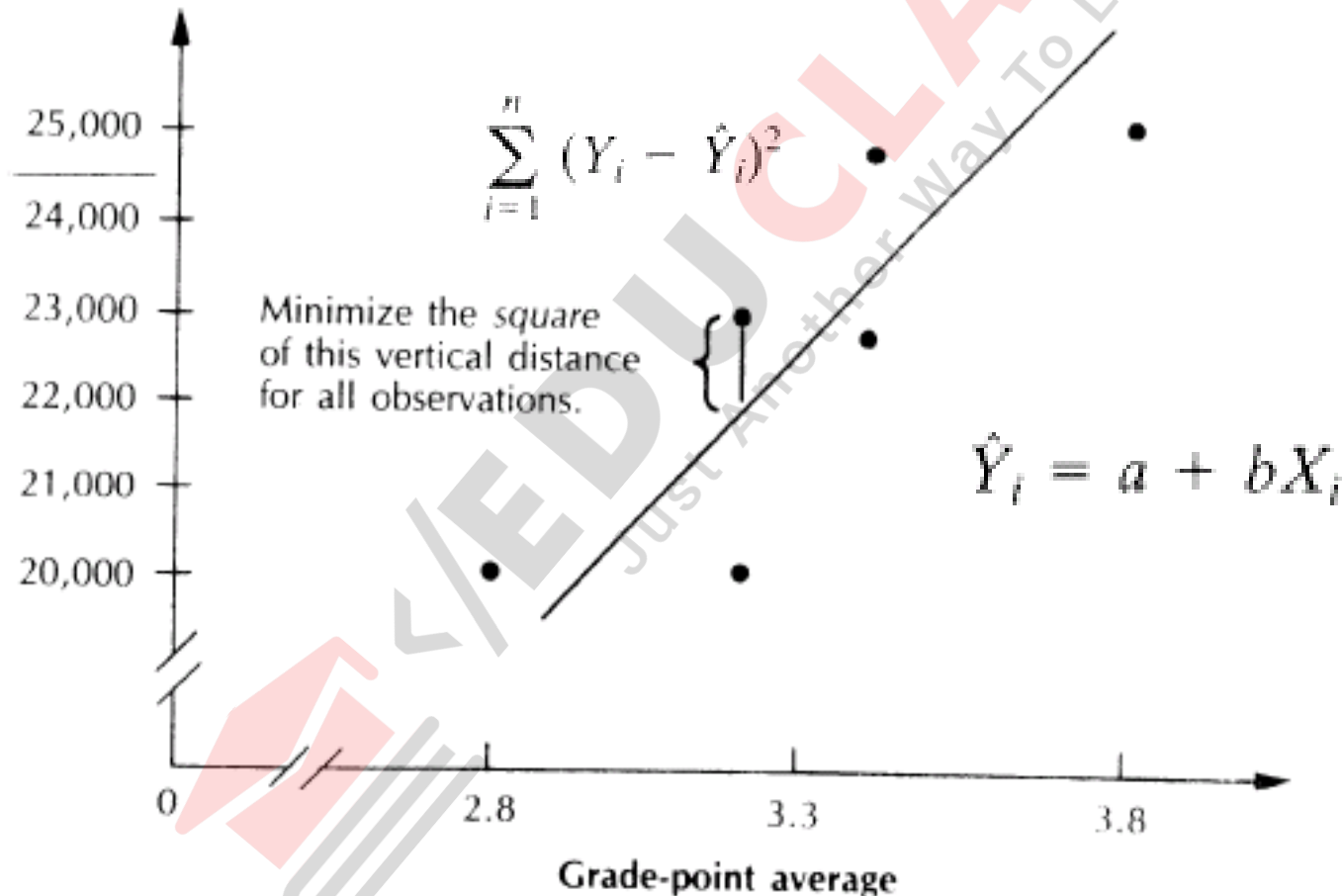
Simple Regression Model

- Find a linear relationship between GPA and starting salary
- As GPA increases/decreases starting salary increases/decreases



Simple Regression Model

- Least Squares Method to find regression model
 - Choose a and b in regression model (equation) so that it minimizes the sum of the squared deviations – actual Y value minus predicted Y value (Y-hat)



Simple Regression Model

- How good is the model?

$$\hat{Y} = 4,779 + 5,370X$$

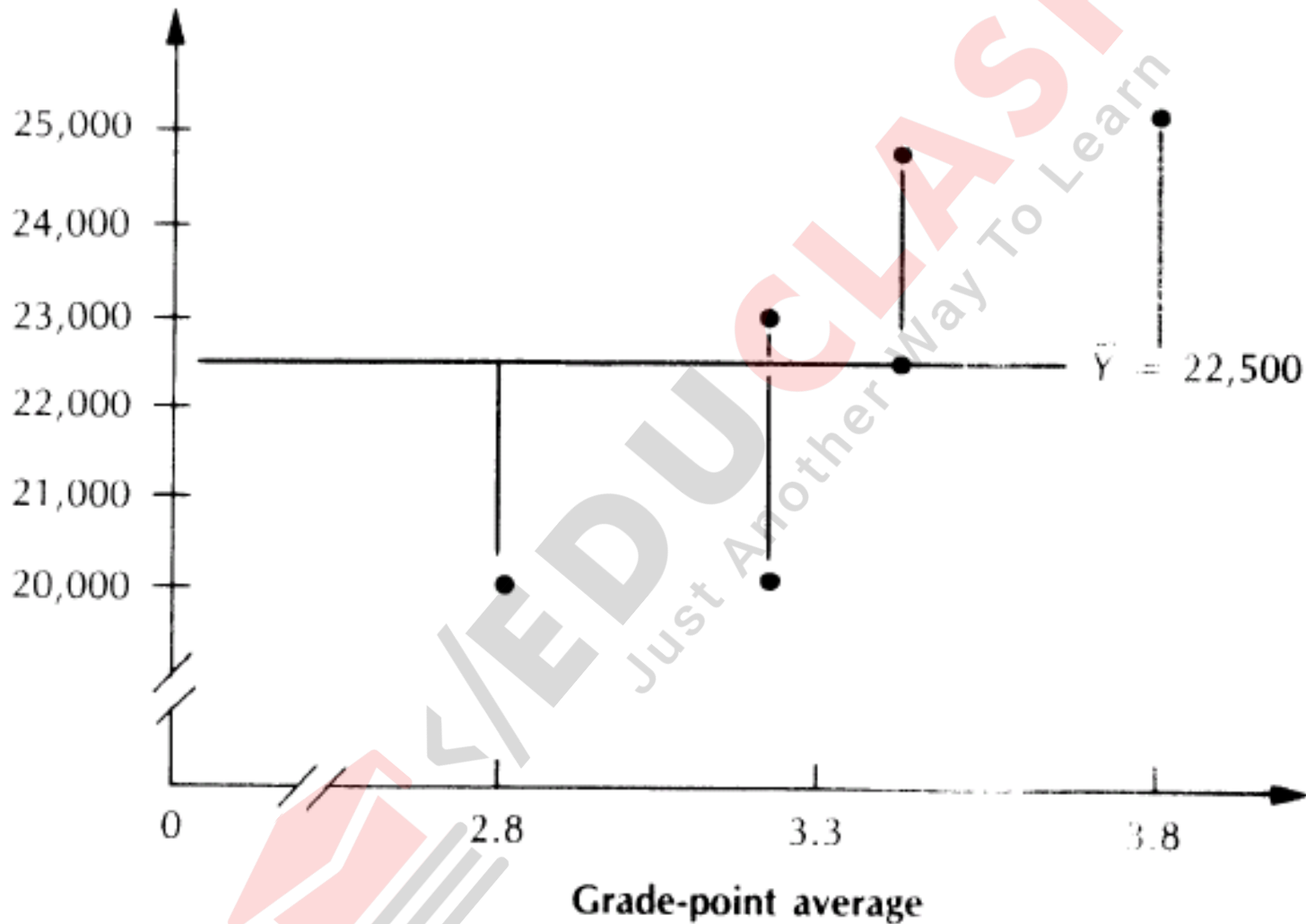
a = 4,779 & b = 5,370

A computer program computed these values

X_i	\hat{Y}_i	Y_i	\hat{Y}_i	\hat{u}_i	\hat{u}_i^2
2.8	19,815	20,000	19,815	185	34,225
3.4	23,037	24,500	23,037	1,463	2,140,369
3.2	21,963	23,000	21,963	1,037	1,075,369
3.8	25,185	25,000	25,185	-185	34,225
3.2	21,963	20,000	21,963	-1,963	3,853,369
3.4	23,037	22,500	23,037	-537	288,369
				<u>0</u>	<u>7,425,926</u>

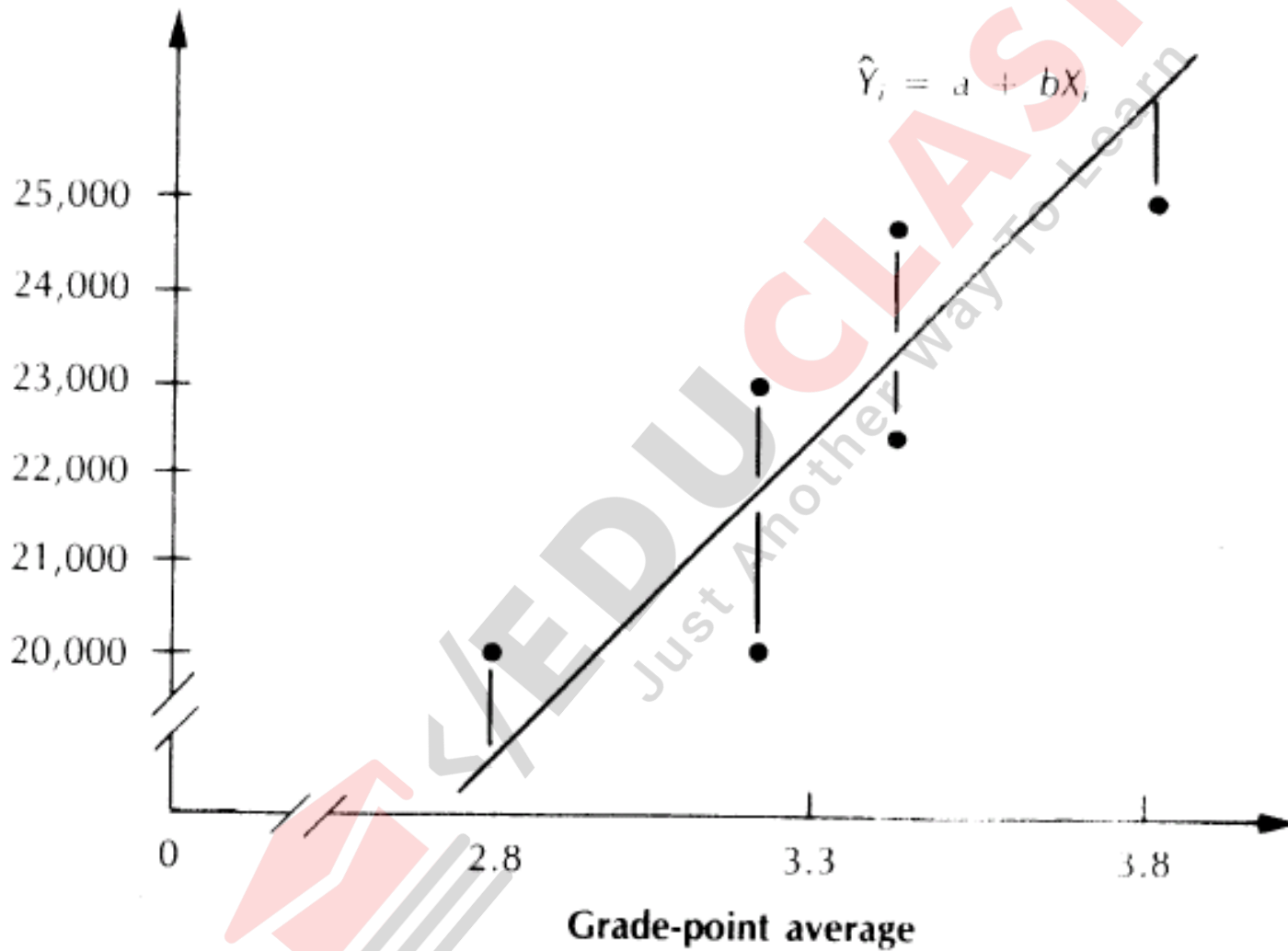
- u-hat is a “**residual**” value
- The sum of all u-hats is zero
- The sum of all u-hats squared is the total variance not explained by the model
- “**unexplained variance**” is 7,425,926

Simple Regression Model



Total Variation = 23,000,000

Simple Regression Model



Total Unexplained Variation = 7,425,726

Simple Regression Model

- Relative Goodness of Fit
 - Summarize the improvement in prediction using regression model
- Compute R^2 – coefficient of determination

$$R^2 = 1 - \frac{\text{Unexplained variation in } Y}{\text{Total variation in } Y}$$

$$R^2 = 1 - \frac{7,425,926}{23,000,000} = 1 - 0.323 = 0.677$$

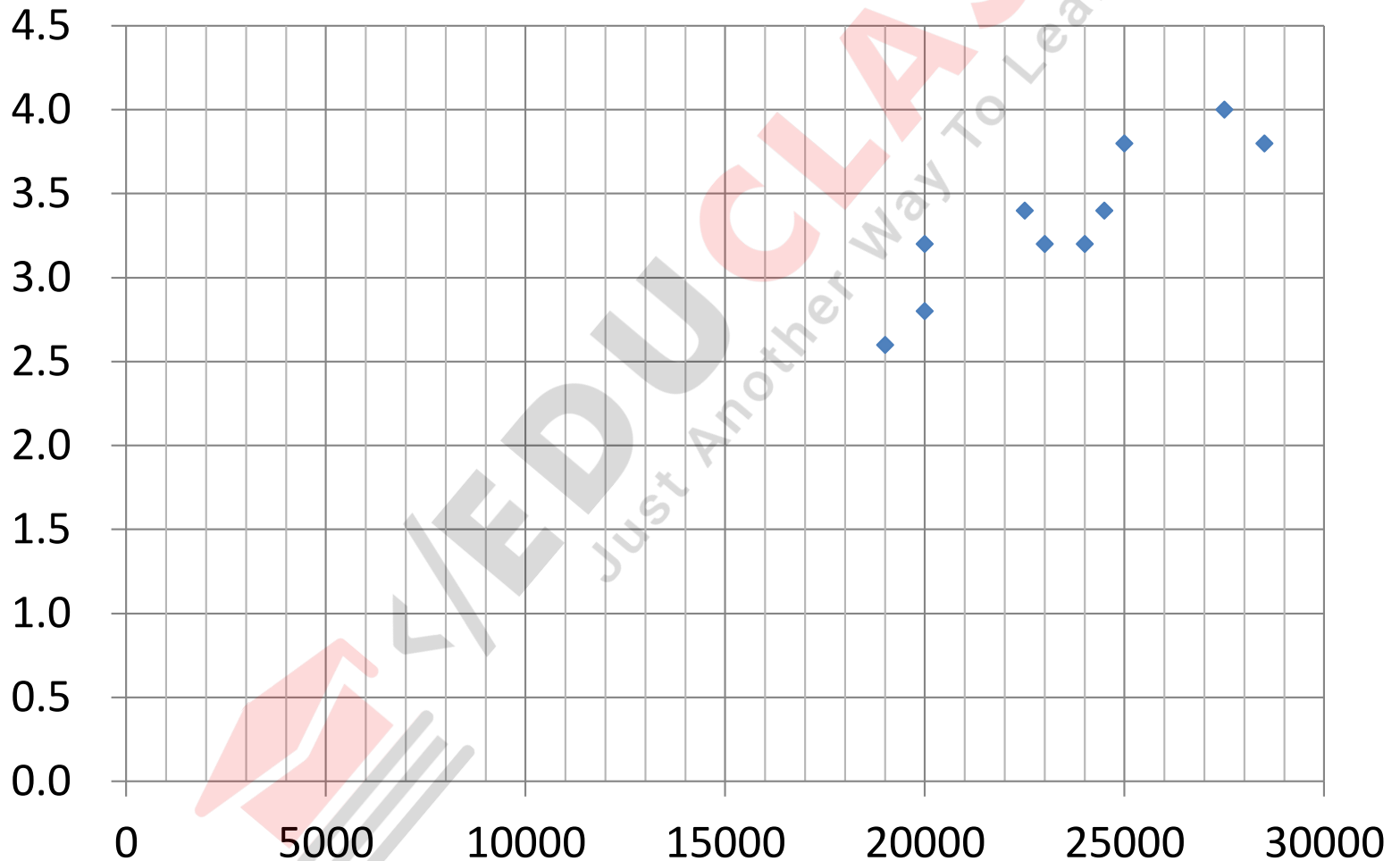
Regression Model (equation) a better predictor than guessing the average salary
The GPA is a more accurate predictor of starting salary than guessing the average
 R^2 is the “performance measure” for the model.

Predicted Starting Salary = $4,779 + 5,370 * \text{GPA}$

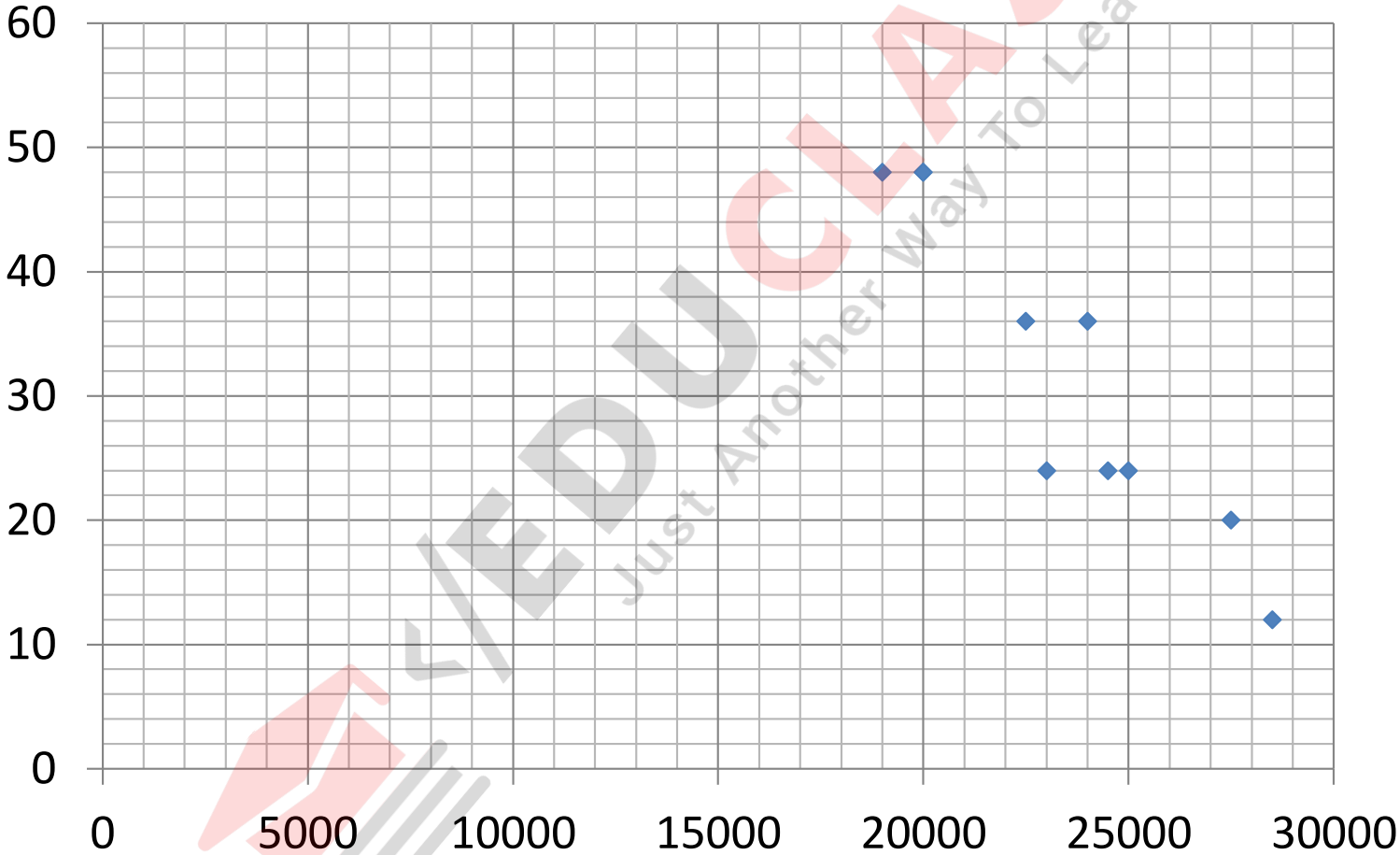
Data Set

Obs #	Salary	GPA	Months Work
1	20000	2.8	48
2	24500	3.4	24
3	23000	3.2	24
4	25000	3.8	24
5	20000	3.2	48
6	22500	3.4	36
7	27500	4.0	20
8	19000	2.6	48
9	24000	3.2	36
10	28500	3.8	12

Scatter Plot - GPA vs Salary



Scatter Plot - Work vs Salary



Three Regressions

- Salary = f(GPA)
- Salary = f(Work)
- Salary = f(GPA, Work)
- Interpret Excel Output



Interpreting Results

- Regression Statistics
 - Multiple R,
 - R^2 ,
 - R^2_{adj}
 - Standard Error S_y
- Statistical Significance
 - t-test
 - p-value
 - F test

Regression Statistics Table

- Multiple R
 - $R = \text{square root of } R^2$
- R^2
 - Coefficient of Determination
- R^2_{adj}
 - used if more than one x variable
- Standard Error S_y
 - This is the sample estimate of the standard deviation of the error (actual – predicted)

Regression Coefficients Table

- Coefficients Column gives
 - $b_0, b_1, b_2, \dots, b_n$ values for the regression equation.
 - The b_0 is the intercept
 - b_1 value is next to your independent variable x_1
 - b_2 is next to your independent variable x_2 .
 - b_3 is next to your independent variable x_3



Regression Coefficients Table

- p values for individual t tests each independent variables
- *t test* - tests the claim that there is no relationship between the independent variable (in the corresponding row) and your dependent variable.
- Should reject the claim
 - *Of **NO** significant relationship between your independent variable (in the corresponding row) and dependent variable if $p < \alpha$.*



Salary = f(GPA)

<i>Regression Statistics</i>	<i>f(GPA)</i>
Multiple R	0.898006642
R Square	0.806415929
Adjusted R Square	0.78221792
Standard Error	1479.019946
Observations	10

ANOVA

	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>
Regression	1	72900000	72900000	33.32571	0.00041792
Residual	8	17500000	2187500		
Total	9	90400000			

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>
Intercept	1928.571429	3748.677	0.514467	0.620833	-6715.89326	10573.04
GPA	6428.571429	1113.589	5.772843	0.000418	3860.63173	8996.511

Salary = f(Work)

<i>Regression Statistics</i>	<i>f(Work)</i>
Multiple R	0.939265177
R Square	0.882219073
Adjusted R Square	0.867496457
Standard Error	1153.657002
Observations	10

ANOVA					
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>
Regression	1	79752604.17	79752604	59.92271	5.52993E-05
Residual	8	10647395.83	1330924		
Total	9	90400000			

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>
Intercept	30691.66667	1010.136344	30.38369	1.49E-09	28362.28808	33021.0453
Months Work	-227.864583	29.43615619	-7.74098	5.53E-05	295.7444812	-159.98469

Salary = f(GPA, Work)

<i>Regression Statistics</i>	<i>f(GPA,Work)</i>
Multiple R	0.962978985
R Square	0.927328525
Adjusted R Square	0.906565246
Standard Error	968.7621974
Observations	10

ANOVA					
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>
Regression	2	83830499	41915249	44.66195	0.00010346
Residual	7	6569501	938500.2		
Total	9	90400000			

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>
Intercept	19135.92896	5608.184	3.412144	0.011255	5874.682112	32397.176
GPA	2725.409836	1307.468	2.084495	0.075582	-366.2602983	5817.08
Months						-
Work	-151.2124317	44.30826	-3.41274	0.011246	-255.9848174	46.440046

Compare Three “Models”

<i>Regression Statistics</i>	<i>f(GPA)</i>
Multiple R	0.898006642
R Square	0.806415929
Adjusted R Square	0.78221792
Standard Error	1479.019946
Observations	10

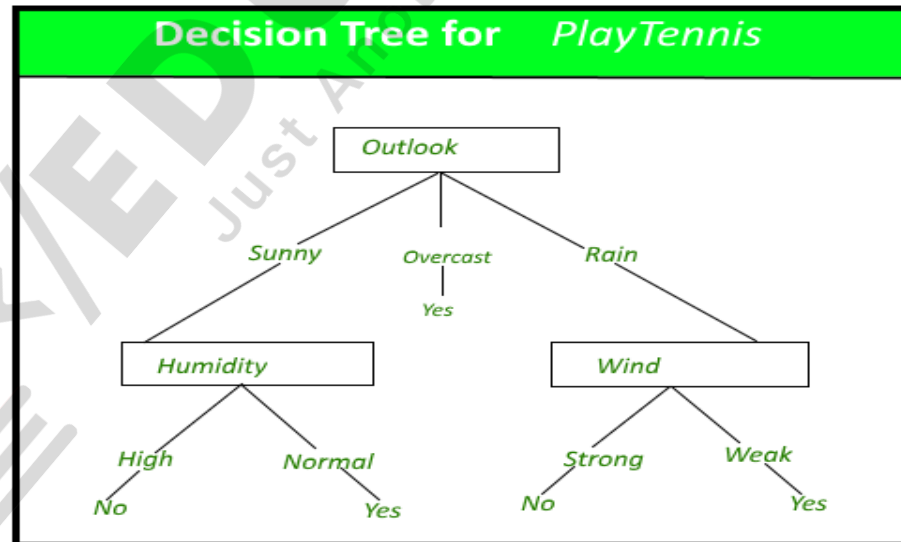
<i>Regression Statistics</i>	<i>f(Work)</i>
Multiple R	0.939265177
R Square	0.882219073
Adjusted R Square	0.867496457
Standard Error	1153.657002
Observations	10

<i>Regression Statistics</i>	<i>f(GPA,Work)</i>
Multiple R	0.962978985
R Square	0.927328525
Adjusted R Square	0.906565246
Standard Error	968.7621974
Observations	10

Decision Tree

- Decision tree algorithm falls under the category of supervised learning. Decision tree is the most powerful and popular tool for classification and prediction.

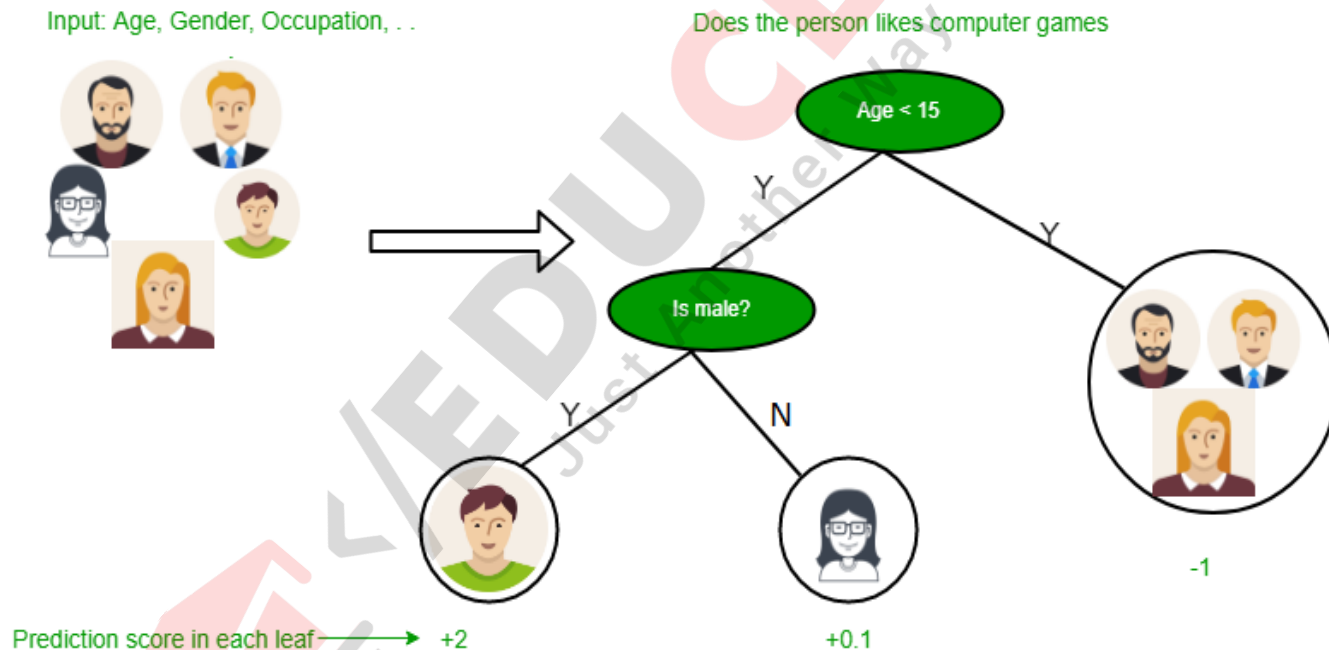
- A Decision tree is a hierarchical structure that represents a test or representation of a data set. Each internal node is a test on a feature, and each leaf node is a predicted class label.



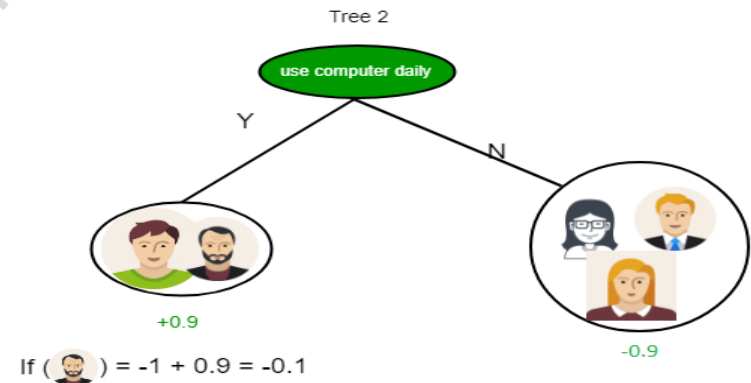
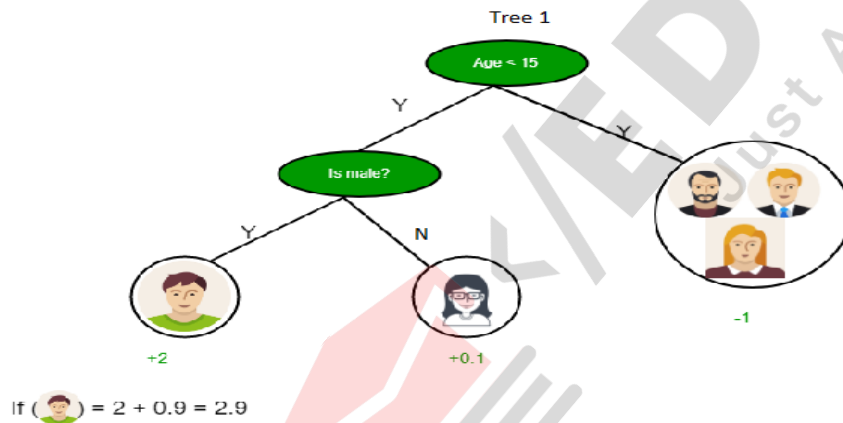
- Decision trees classify instances by sorting them down the tree from the root to some leaf node, which provides the classification of the instance.
- An instance is classified by starting at the root node of the tree, testing the attribute specified by this node, then moving down the tree branch corresponding to the value of the attribute as shown in the above figure.

- The decision tree in above figure classifies a particular morning according to whether it is suitable for playing tennis and returning the classification associated with the particular leaf.(in this case Yes or No). For example,
- *(Outlook = Rain, Temperature = Hot, Humidity = High, Wind = Strong)*
- the instance would be sorted down the leftmost branch of this decision tree and

- We can represent any boolean function on discrete attributes using the decision tree.



- **Below are some assumptions that we made while using decision tree:**
- At the beginning, we consider the whole training set as the root.
- Feature values are preferred to be categorical. If the values are continuous then they are discretized prior to building the model.
- On the basis of attribute values records are distributed recursively.
- We use statistical methods for ordering attributes as root or the internal node.



- In Decision Tree the major challenge is to identification of the attribute for the root node in each level. This process is known as attribute selection.
- We have two popular attribute selection measures:
 1. Information Gain
 2. Gini Index

ID3

- ID3 stands for Iterative Dichotomiser 3
- It is a classification algorithm that follows a greedy approach by selecting a best attribute that yields maximum Information Gain(IG) or minimum Entropy(H).
- **Entropy**
- Entropy is a measure of the amount of uncertainty in the dataset S. Mathematical Representation of Entropy is shown here -
- $H(S) = -\sum_{c \in C} p(c) \log_2 p(c)$ Where,
- S - The current dataset for which entropy is being calculated (changes every iteration of the ID3 algorithm).
- C - Set of classes in S {example - C = {yes, no}}
- p(c) - The proportion of the number of elements in class c to the number of elements in set S.

- In ID3, entropy is calculated for each remaining attribute. The attribute with the smallest entropy is used to split the set S on that particular iteration. Entropy = 0 implies it is of pure class, that means all are of same category.
 - **Information Gain IG(A) tells us how much uncertainty in S was reduced after splitting set S on attribute A. Mathematical representation of Information gain is shown here -**
 - $IG(A,S)=H(S)-\sum_{t \in T} p(t)H(t)$
- Where,
- H(S) - Entropy of set S.
 - T - The subsets created from splitting set S by attribute A such that $S=\bigcup_{t \in T} t$
 - p(t) - The proportion of the number of elements in t to the number of elements in set S.
 - H(t) - Entropy of subset t.
 - In ID3, information gain can be calculated (instead of entropy) for each remaining attribute. The attribute with the largest information gain is used to split the set S on that particular iteration.

ID3 Algorithm

- Calculate entropy for dataset.
- For each attribute/feature
 - Calculate entropy for all its categorical values.
 - Calculate information gain for the feature.
- Find the feature with maximum information gain.
- Repeat it until we get the desired tree.

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

- Here, dataset is of binary classes (yes and no), where 9 out of 14 are "yes" and 5 out of 14 are "no".

- Complete entropy of dataset is -

$$H(S) = - p(\text{yes}) * \log_2(p(\text{yes})) - p(\text{no}) * \log_2(p(\text{no}))$$

$$= - (9/14) * \log_2(9/14) - (5/14) * \log_2(5/14)$$

$$= - (-0.41) - (-0.53)$$

First Attribute - Outlook

Categorical values - sunny, overcast and rain

$$H(\text{Outlook}=\text{sunny}) = - (2/5) * \log(2/5) - (3/5) * \log(3/5) \\ = 0.971$$

$$H(\text{Outlook}=\text{rain}) = - (3/5) * \log(3/5) - (2/5) * \log(2/5) \\ = 0.971$$

$$H(\text{Outlook}=\text{overcast}) = - (4/4) * \log(4/4) - 0 \\ = 0$$

Average Entropy Information for Outlook -

$$I(\text{Outlook}) = p(\text{sunny}) * H(\text{Outlook}=\text{sunny}) + p(\text{rain}) * H(\text{Outlook}=\text{rain}) \\ + p(\text{overcast}) * H(\text{Outlook}=\text{overcast}) \\ = (5/14) * 0.971 + (5/14) * 0.971 + (4/14) * 0 \\ = 0.693$$

$$\text{Information Gain} = H(S) - I(\text{Outlook}) \\ = 0.94 - 0.693 \\ = 0.247$$

Second Attribute - Temperature

Categorical values - hot, mild, cool H

$$H(\text{Temperature}=\text{hot}) = - (2/4) * \log (2/4) - (2/4) * \log (2/4) = 1$$

$$H(\text{Temperature}=\text{cool}) = - (3/4) * \log (3/4) - (1/4) * \log (1/4) = 0.811$$

$$H(\text{Temperature}=\text{mild}) = - (4/6) * \log (4/6) - (2/6) * \log (2/6) = 0.9179$$

Average Entropy Information for Temperature

$$I(\text{Temperature}) = p(\text{hot}) * H(\text{Temperature}=\text{hot}) +$$

$$p(\text{mild}) * H(\text{Temperature}=\text{mild}) + p(\text{cool}) * H(\text{Temperature}=\text{cool}) \\ = (4/14) * 1 + (6/14) * 0.9179 + (4/14) * 0.811 = 0.9108 \text{ Information}$$

$$\text{Gain} = H(S) - I(\text{Temperature}) \\ = 0.94 - 0.9108 = 0.0292$$

Third Attribute - Humidity

Categorical values - high, normal

$$H(\text{Humidity}=\text{high}) = -(3/7) * \log(3/7) - (4/7) * \log(4/7) = 0.983$$

$$H(\text{Humidity}=\text{normal}) = -(6/7) * \log(6/7) - (1/7) * \log(1/7) = 0.591$$

Average Entropy Information for Humidity

$$\begin{aligned} I(\text{Humidity}) &= p(\text{high}) * H(\text{Humidity}=\text{high}) + p(\text{normal}) * H(\text{Humidity}=\text{normal}) \\ &= (7/14) * 0.983 + (7/14) * 0.591 \\ &= 0.787 \end{aligned}$$

$$\begin{aligned} \text{Information Gain} &= H(S) - I(\text{Humidity}) \\ &= 0.94 - 0.787 = 0.153 \end{aligned}$$

Fourth Attribute - Wind

Categorical values - weak, strong

$$H(\text{Wind}=\text{weak}) = -(6/8) * \log(6/8) - (2/8) * \log(2/8) = 0.811$$

$$H(\text{Wind}=\text{strong}) = -(3/6) * \log(3/6) - (3/6) * \log(3/6) = 1$$

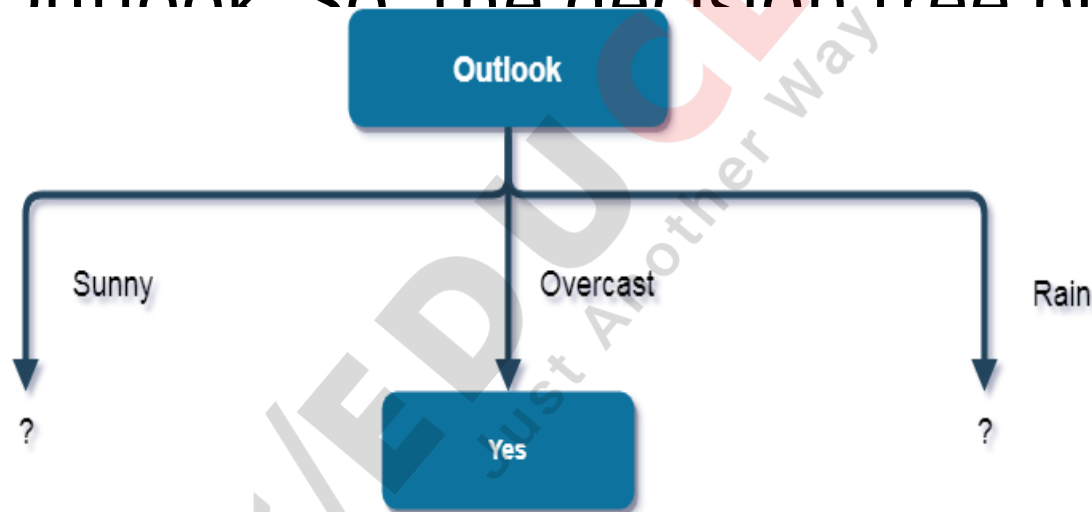
Average Entropy Information for Wind

$$\begin{aligned} I(\text{Wind}) &= p(\text{weak}) * H(\text{Wind}=\text{weak}) + p(\text{strong}) * H(\text{Wind}=\text{strong}) \\ &= (8/14) * 0.811 + (6/14) * 1 = 0.892 \end{aligned}$$

$$\begin{aligned} \text{Information Gain} &= H(S) - I(\text{Wind}) \\ &= 0.94 - 0.892 = 0.048 \end{aligned}$$

Cont..

- Here, the attribute with maximum information gain is Outlook. So, the decision tree built so far



- Now, finding the best attribute for splitting the dataset rows

Complete entropy of Sunny is

$$\begin{aligned} H(S) &= -p(\text{yes}) * \log_2(p(\text{yes})) - p(\text{no}) * \log_2(p(\text{no})) \\ &= - (2/5) * \log_2(2/5) - (3/5) * \log_2(3/5) \\ &= 0.971 \end{aligned}$$



First Attribute - Temperature

Categorical values - hot, mild, cool

$$H(\text{Sunny}, \text{Temperature}=\text{hot}) = -0 - (2/2) * \log(2/2) = 0$$

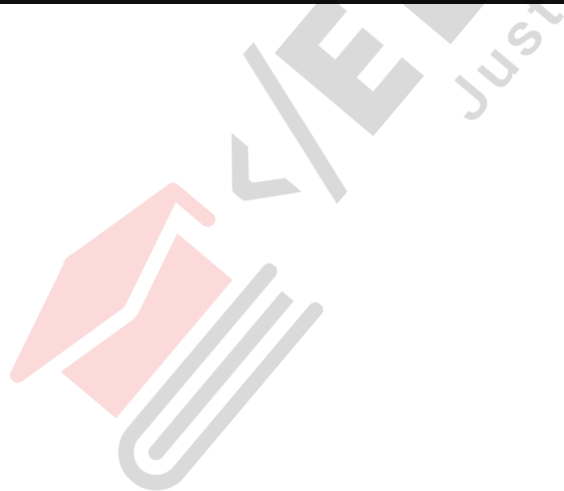
$$H(\text{Sunny}, \text{Temperature}=\text{cool}) = -(1) * \log(1) - 0 = 0$$

$$H(\text{Sunny}, \text{Temperature}=\text{mild}) = -(1/2) * \log(1/2) - (1/2) * \log(1/2) = 1$$

Average Entropy Information for Temperature

$$\begin{aligned} I(\text{Sunny}, \text{Temperature}) &= p(\text{Sunny}, \text{hot}) * H(\text{Sunny}, \text{Temperature}=\text{hot}) + p(\text{Sunny}, \\ \text{mild}) * H(\text{Sunny}, \text{Temperature}=\text{mild}) + p(\text{Sunny}, \text{cool}) * H(\text{Sunny}, \text{Temperature}=\text{cool}) \\ &= (2/5) * 0 + (1/5) * 0 + (2/5) * 1 = 0.4 \end{aligned}$$

$$\begin{aligned} \text{Information Gain} &= H(\text{Sunny}) - I(\text{Sunny}, \text{Temperature}) \\ &= 0.971 - 0.4 = 0.571 \end{aligned}$$



Second Attribute - Humidity

Categorical values - high, normal

$$H(\text{Sunny}, \text{Humidity}=\text{high}) = -0 - (3/3) * \log(3/3) = 0$$

$$H(\text{Sunny}, \text{Humidity}=\text{normal}) = -(2/2) * \log(2/2) - 0 = 0$$

Average Entropy Information for Humidity

$$I(\text{Sunny}, \text{Humidity}) = p(\text{Sunny}, \text{high}) * H(\text{Sunny}, \text{Humidity}=\text{high}) + p(\text{Sunny}, \text{normal}) * H(\text{Sunny}, \text{Humidity}=\text{normal})$$

$$= (3/5) * 0 + (2/5) * 0 = 0$$

$$\text{Information Gain} = H(\text{Sunny}) - I(\text{Sunny}, \text{Humidity})$$

$$= 0.971 - 0 = 0.971$$



Third Attribute - Wind

Categorical values - weak, strong

$$H(\text{Sunny}, \text{Wind}=\text{weak}) = -(1/3) \cdot \log(1/3) - (2/3) \cdot \log(2/3) = 0.918$$

$$H(\text{Sunny}, \text{Wind}=\text{strong}) = -(1/2) \cdot \log(1/2) - (1/2) \cdot \log(1/2) = 1$$

Average Entropy Information for Wind

$$I(\text{Sunny}, \text{Wind}) = p(\text{Sunny}, \text{weak}) \cdot H(\text{Sunny}, \text{Wind}=\text{weak}) + p(\text{Sunny}, \text{strong}) \cdot H(\text{Sunny}, \text{Wind}=\text{strong})$$

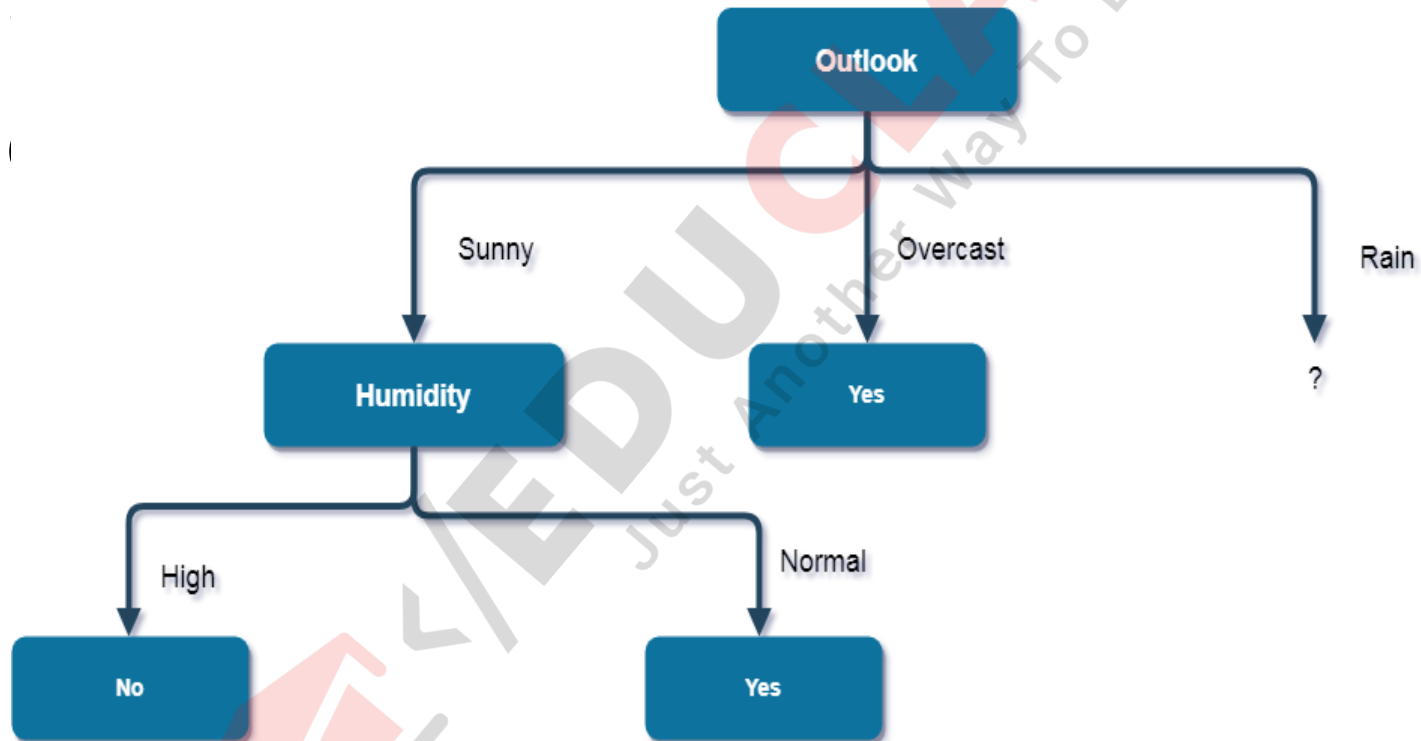
$$= (3/5) \cdot 0.918 + (2/5) \cdot 1 = 0.9508$$

$$\text{Information Gain} = H(\text{Sunny}) - I(\text{Sunny}, \text{Wind})$$

$$= 0.971 - 0.9508$$

$$= 0.0202$$

- Here, the attribute with maximum



- Here, when Outlook = Sunny and Humidity = High, it is a pure class of category "no". And When Outlook = Sunny and Humidity = Normal, it is again a pure class of category

Complete entropy of Rain is

$$\begin{aligned} H(S) &= - p(\text{yes}) * \log_2(p(\text{yes})) - p(\text{no}) * \log_2(p(\text{no})) \\ &= - (3/5) * \log_2(3/5) - (2/5) * \log_2(2/5) \\ &= 0.971 \end{aligned}$$

- Now, finding the best attribute for splitting the data with Outlook=Sunny values{ Dataset rows = [4, 5, 6, 10, 14]}.

First Attribute - Temperature

Categorical values - mild, cool

$$H(\text{Rain}, \text{Temperature}=\text{cool}) = -(1/2) * \log(1/2) - (1/2) * \log(1/2) = 1$$

$$H(\text{Rain}, \text{Temperature}=\text{mild}) = -(2/3) * \log(2/3) - (1/3) * \log(1/3) = 0.918$$

Average Entropy Information for Temperature

$$\begin{aligned} I(\text{Rain}, \text{Temperature}) &= p(\text{Rain}, \text{mild}) * H(\text{Rain}, \text{Temperature}=\text{mild}) + \\ & p(\text{Rain}, \text{cool}) * H(\text{Rain}, \text{Temperature}=\text{cool}) \\ &= (2/5) * 1 + (3/5) * 0.918 \\ &= 0.9508 \end{aligned}$$

$$\begin{aligned} \text{Information Gain} &= H(\text{Rain}) - I(\text{Rain}, \text{Temperature}) \\ &= 0.971 - 0.9508 \\ &= 0.0202 \end{aligned}$$



Second Attribute - Wind

Categorical values - weak, strong

$$H(\text{Wind}=\text{weak}) = -\left(\frac{3}{3}\right) \log\left(\frac{3}{3}\right) - 0 = 0$$

$$H(\text{Wind}=\text{strong}) = 0 - \left(\frac{2}{2}\right) \log\left(\frac{2}{2}\right) = 0$$

Average Entropy Information for Wind

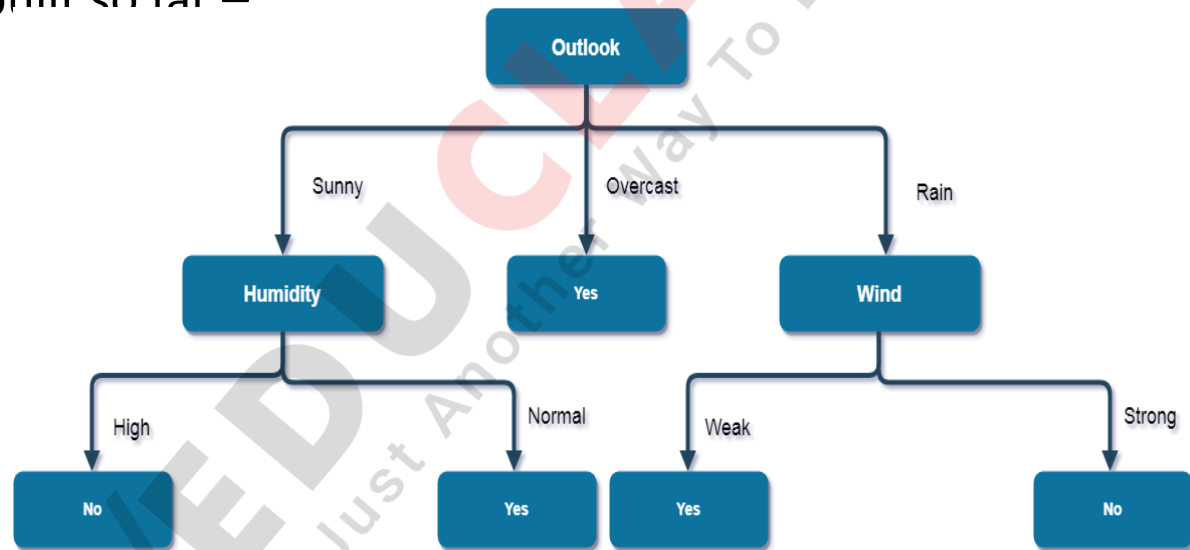
$$I(\text{Wind}) = p(\text{Rain}, \text{weak}) * H(\text{Rain}, \text{Wind}=\text{weak}) + p(\text{Rain}, \text{strong}) * H(\text{Rain}, \text{Wind}=\text{strong})$$

$$= \left(\frac{3}{5}\right) * 0 + \left(\frac{2}{5}\right) * 0 = 0$$

$$\text{Information Gain} = H(\text{Rain}) - I(\text{Rain}, \text{Wind})$$

$$= 0.971 - 0 = 0.971$$

- Here, the attribute with maximum information gain is Wind. So, the decision tree built so far –



- Here, when Outlook = Rain and Wind = Strong, it is a pure class or category "no". And When Outlook = Rain and Wind = Weak, it is again a pure class of category "yes".
And this is our final desired tree for the given dataset.

Characteristics of ID3 algorithm

- ID3 uses a greedy approach that's why it does not guarantee an optimal solution; it can get stuck in local optimums.
- ID3 can overfit to the training data (to avoid overfitting, smaller decision trees should be preferred over larger ones).
- This algorithm usually produces small trees, but it does not always produce the smallest possible tree.
- ID3 is harder to use on continuous data (if

C4.5

- Firstly, we need to calculate global entropy. There are 14 examples; 9 instances refer to yes decision, and 5 instances refer to no decision.
- $\text{Entropy}(\text{Decision}) = \sum -p(l) \cdot \log_2 p(l) = -p(\text{Yes}) \cdot \log_2 p(\text{Yes}) - p(\text{No}) \cdot \log_2 p(\text{No}) = -(9/14) \cdot \log_2(9/14) - (5/14) \cdot \log_2(5/14) = 0.940$
- In ID3 algorithm, we've calculated gains for each attribute. Here, we need to calculate

Wind Attribute

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	85	85	Weak	No
2	Sunny	80	90	Strong	No
3	Overcast	83	78	Weak	Yes
4	Rain	70	96	Weak	Yes
5	Rain	68	80	Weak	Yes
6	Rain	65	70	Strong	No
7	Overcast	64	65	Strong	Yes
8	Sunny	72	95	Weak	No
9	Sunny	69	70	Weak	Yes
10	Rain	75	80	Weak	Yes
11	Sunny	75	70	Strong	Yes
12	Overcast	72	90	Strong	Yes
13	Overcast	81	75	Weak	Yes
14	Rain	71	80	Strong	No

Wind Attribute

- Wind is a nominal attribute. Its possible values are weak and strong.
- $\text{Gain}(\text{Decision}, \text{Wind}) = \text{Entropy}(\text{Decision}) - \sum (p(\text{Decision} | \text{Wind}) \cdot \text{Entropy}(\text{Decision} | \text{Wind}))$
- $\text{Gain}(\text{Decision}, \text{Wind}) = \text{Entropy}(\text{Decision}) - [p(\text{Decision} | \text{Wind}=\text{Weak}) \cdot \text{Entropy}(\text{Decision} | \text{Wind}=\text{Weak})] + [p(\text{Decision} | \text{Wind}=\text{Strong}) \cdot \text{Entropy}(\text{Decision} | \text{Wind}=\text{Strong})]$
- There are 8 weak wind instances. 2 of them are concluded as no, 6 of them are concluded as yes.
- $\text{Entropy}(\text{Decision} | \text{Wind}=\text{Weak}) = - p(\text{No}) \cdot \log_2 p(\text{No}) - p(\text{Yes}) \cdot \log_2 p(\text{Yes}) = - (2/8) \cdot \log_2(2/8) - (6/8) \cdot \log_2(6/8) = 0.811$
- $\text{Entropy}(\text{Decision} | \text{Wind}=\text{Strong}) = - (3/6) \cdot \log_2(3/6) - (3/6) \cdot \log_2(3/6) = 1$
- $\text{Gain}(\text{Decision}, \text{Wind}) = 0.940 - (8/14) \cdot (0.811) - (6/14) \cdot (1) = 0.940 - 0.463 - 0.428 = 0.049$
- There are 8 decisions for weak wind, and 6 decisions for strong wind.
- $\text{SplitInfo}(\text{Decision}, \text{Wind}) = -(8/14) \cdot \log_2(8/14) - (6/14) \cdot \log_2(6/14) = 0.461 + 0.524 = 0.985$
- $\text{GainRatio}(\text{Decision}, \text{Wind}) = \text{Gain}(\text{Decision}, \text{Wind}) / \text{SplitInfo}(\text{Decision}, \text{Wind}) = 0.049 / 0.985 = 0.049$

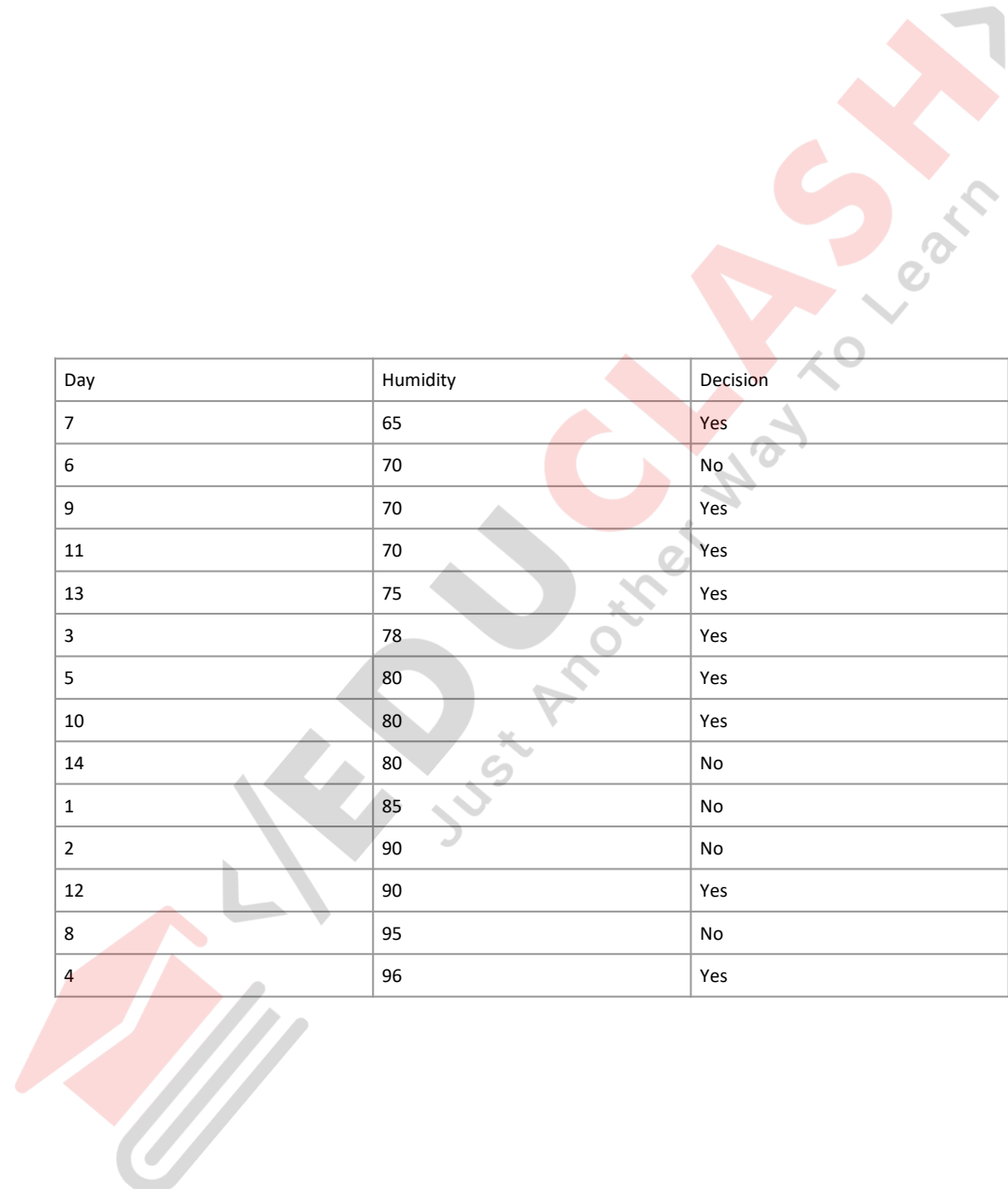


Outlook Attribute

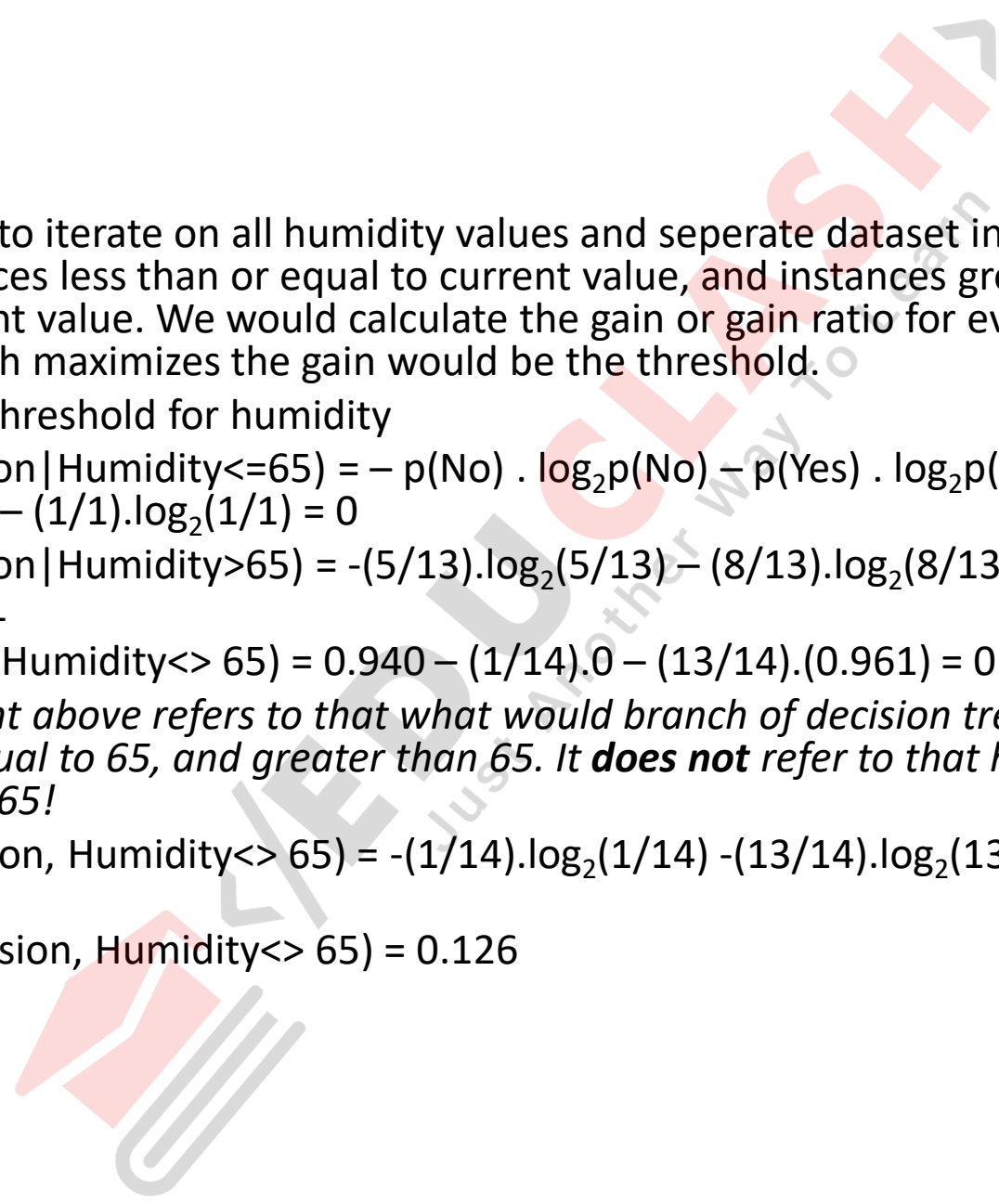
- Outlook is a nominal attribute, too. Its possible values are sunny, overcast and rain.
- $\text{Gain}(\text{Decision}, \text{Outlook}) = \text{Entropy}(\text{Decision}) - \sum (p(\text{Decision} | \text{Outlook}) \cdot \text{Entropy}(\text{Decision} | \text{Outlook})) =$
- $\text{Gain}(\text{Decision}, \text{Outlook}) = \text{Entropy}(\text{Decision}) - p(\text{Decision} | \text{Outlook}=\text{Sunny}) \cdot \text{Entropy}(\text{Decision} | \text{Outlook}=\text{Sunny}) - p(\text{Decision} | \text{Outlook}=\text{Overcast}) \cdot \text{Entropy}(\text{Decision} | \text{Outlook}=\text{Overcast}) - p(\text{Decision} | \text{Outlook}=\text{Rain}) \cdot \text{Entropy}(\text{Decision} | \text{Outlook}=\text{Rain})$
- There are 5 sunny instances. 3 of them are concluded as no, 2 of them are concluded as yes.
- $\text{Entropy}(\text{Decision} | \text{Outlook}=\text{Sunny}) = - p(\text{No}) \cdot \log_2 p(\text{No}) - p(\text{Yes}) \cdot \log_2 p(\text{Yes}) = -(3/5) \cdot \log_2(3/5) - (2/5) \cdot \log_2(2/5) = 0.441 + 0.528 = 0.970$
- $\text{Entropy}(\text{Decision} | \text{Outlook}=\text{Overcast}) = - p(\text{No}) \cdot \log_2 p(\text{No}) - p(\text{Yes}) \cdot \log_2 p(\text{Yes}) = -(0/4) \cdot \log_2(0/4) - (4/4) \cdot \log_2(4/4) = 0$
- $\text{Entropy}(\text{Decision} | \text{Outlook}=\text{Rain}) = - p(\text{No}) \cdot \log_2 p(\text{No}) - p(\text{Yes}) \cdot \log_2 p(\text{Yes}) = -(2/5) \cdot \log_2(2/5) - (3/5) \cdot \log_2(3/5) = 0.528 + 0.441 = 0.970$
- $\text{Gain}(\text{Decision}, \text{Outlook}) = 0.940 - (5/14) \cdot (0.970) - (4/14) \cdot (0) - (5/14) \cdot (0.970) - (5/14) \cdot (0.970) = 0.246$
- There are 5 instances for sunny, 4 instances for overcast and 5 instances for rain
- $\text{SplitInfo}(\text{Decision}, \text{Outlook}) = -(5/14) \cdot \log_2(5/14) - (4/14) \cdot \log_2(4/14) - (5/14) \cdot \log_2(5/14) = 1.577$
- $\text{GainRatio}(\text{Decision}, \text{Outlook}) = \text{Gain}(\text{Decision}, \text{Outlook}) / \text{SplitInfo}(\text{Decision}, \text{Outlook}) = 0.246 / 1.577 = 0.155$

Humidity Attribute

- Humidity is a continuous attribute.
- We need to convert continuous values to nominal ones. C4.5 proposes to perform binary split based on a threshold value.
- Threshold should be a value which offers maximum gain for that attribute. Let's focus on humidity attribute.
- Firstly, we need to sort humidity values smallest to largest.



Day	Humidity	Decision
7	65	Yes
6	70	No
9	70	Yes
11	70	Yes
13	75	Yes
3	78	Yes
5	80	Yes
10	80	Yes
14	80	No
1	85	No
2	90	No
12	90	Yes
8	95	No
4	96	Yes



- Now, we need to iterate on all humidity values and separate dataset into two parts as instances less than or equal to current value, and instances greater than the current value. We would calculate the gain or gain ratio for every step. The value which maximizes the gain would be the threshold.
- Check 65 as a threshold for humidity
- $\text{Entropy}(\text{Decision} | \text{Humidity} \leq 65) = -p(\text{No}) \cdot \log_2 p(\text{No}) - p(\text{Yes}) \cdot \log_2 p(\text{Yes}) = -(0/1) \cdot \log_2(0/1) - (1/1) \cdot \log_2(1/1) = 0$
- $\text{Entropy}(\text{Decision} | \text{Humidity} > 65) = -(5/13) \cdot \log_2(5/13) - (8/13) \cdot \log_2(8/13) = 0.530 + 0.431 = 0.961$
- $\text{Gain}(\text{Decision}, \text{Humidity} \neq 65) = 0.940 - (1/14) \cdot 0 - (13/14) \cdot (0.961) = 0.048$
- ** The statement above refers to that what would branch of decision tree be for less than or equal to 65, and greater than 65. It **does not** refer to that humidity is not equal to 65!*
- $\text{SplitInfo}(\text{Decision}, \text{Humidity} \neq 65) = -(1/14) \cdot \log_2(1/14) - (13/14) \cdot \log_2(13/14) = 0.371$
- $\text{GainRatio}(\text{Decision}, \text{Humidity} \neq 65) = 0.126$

- Check 70 as a threshold for humidity
- Entropy(Decision | Humidity \leq 70) = $-(1/4).\log_2(1/4) - (3/4).\log_2(3/4) = 0.811$
- Entropy(Decision | Humidity $>$ 70) = $-(4/10).\log_2(4/10) - (6/10).\log_2(6/10) = 0.970$
- Gain(Decision, Humidity \neq 70) = $0.940 - (4/14).(0.811) - (10/14).(0.970) = 0.940 - 0.231 - 0.692 = 0.014$
- SplitInfo(Decision, Humidity \neq 70) = $-(4/14).\log_2(4/14) - (10/14).\log_2(10/14) =$

- Check 75 as a threshold for humidity
- $\text{Entropy}(\text{Decision} \mid \text{Humidity} \leq 75) = - (1/5) \cdot \log_2(1/5) - (4/5) \cdot \log_2(4/5) = 0.721$
- $\text{Entropy}(\text{Decision} \mid \text{Humidity} > 75) = - (4/9) \cdot \log_2(4/9) - (5/9) \cdot \log_2(5/9) = 0.991$
- $\text{Gain}(\text{Decision}, \text{Humidity} < > 75) = 0.940 - (5/14) \cdot (0.721) - (9/14) \cdot (0.991) = 0.940 - 0.2575 - 0.637 = 0.045$
- $\text{SplitInfo}(\text{Decision}, \text{Humidity} < > 75) = -(5/14) \cdot \log_2(4/14) - (9/14) \cdot \log_2(10/14) = 0.940$
- $\text{GainRatio}(\text{Decision}, \text{Humidity} < > 75) = 0.047$

- Similarly we can calculate-
- $\text{Gain}(\text{Decision}, \text{Humidity} <> 78) = 0.090$, $\text{GainRatio}(\text{Decision}, \text{Humidity} <> 78) = 0.090$
- **$\text{Gain}(\text{Decision}, \text{Humidity} <> 80) = 0.101$, $\text{GainRatio}(\text{Decision}, \text{Humidity} <> 80) = 0.107$**
- $\text{Gain}(\text{Decision}, \text{Humidity} <> 85) = 0.024$, $\text{GainRatio}(\text{Decision}, \text{Humidity} <> 85) = 0.027$
- $\text{Gain}(\text{Decision}, \text{Humidity} <> 90) = 0.010$, $\text{GainRatio}(\text{Decision}, \text{Humidity} <> 90) = 0.016$
- $\text{Gain}(\text{Decision}, \text{Humidity} <> 95) = 0.048$, $\text{GainRatio}(\text{Decision}, \text{Humidity} <> 95) = 0.128$
- Here, I ignore the value 96 as threshold because humidity cannot be greater than this value.

- As seen, gain maximizes when threshold is equal to 80 for humidity. This means that we need to compare other nominal attributes and comparison of humidity to 80 to create a branch in our tree.

- | Attribute | Gain | GainRatio |
|-------------------|-------|-----------|
| Wind | 0.049 | 0.049 |
| Outlook | 0.246 | 0.155 |
| Humidity <> 80 | 0.101 | 0.107 |
| Temperature <> 83 | 0.113 | 0.305 |

Temperature <> 83 is well. Temperature for Outlook is 0.155. Temperature for Outlook is 0.155. Temperature for Outlook is 0.155.

- C4.5 algorithm solves most of problems in ID3.
- The algorithm uses gain ratios instead of gains.
- In this way, it creates more generalized trees and not to fall into overfitting.
- Moreover, the algorithm transforms continuous attributes to nominal ones based on gain maximization and in this way it can handle continuous data.
- Additionally, it can ignore instances including missing data and handle missing dataset. On the other hand, both ID3 and C4.5 requires high CPU and memory demand.



CART

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

- **Gini index**
- Gini index is a metric for classification tasks in CART. It stores sum of squared probabilities of each class. We can formulate it as illustrated below.
- $Gini = 1 - \sum (P_i)^2$ for $i=1$ to number of classes



Outlook

- Outlook is a nominal feature. It can be sunny, overcast or rain. I will summarize the final decisions for outlook feature.

Outlook	Yes	No	Number of instances
Sunny	2	3	5
Overcast	4	0	4
Rain	3	2	5



- $\text{Gini}(\text{Outlook}=\text{Sunny}) = 1 - (2/5)^2 - (3/5)^2 = 1 - 0.16 - 0.36 = 0.48$
- $\text{Gini}(\text{Outlook}=\text{Overcast}) = 1 - (4/4)^2 - (0/4)^2 = 0$
- $\text{Gini}(\text{Outlook}=\text{Rain}) = 1 - (3/5)^2 - (2/5)^2 = 1 - 0.36 - 0.16 = 0.48$
- Then, we will calculate weighted sum of gini indexes for outlook feature.
- $\text{Gini}(\text{Outlook}) = (5/14) \times 0.48 + (4/14) \times 0 + (5/14) \times 0.48 = 0.171 + 0 + 0.171 = 0.342$

Temperature

- Similarly, temperature is a nominal feature and it could have 3 different values: Cool, Hot and Mild. Let's summarize decisions for

temp

Temperature	Yes	No	Number of instances
Hot	2	2	4
Cool	3	1	4
Mild	4	2	6



- $\text{Gini}(\text{Temp}=\text{Hot}) = 1 - (2/4)^2 - (2/4)^2 = 0.5$
- $\text{Gini}(\text{Temp}=\text{Cool}) = 1 - (3/4)^2 - (1/4)^2 = 1 - 0.5625 - 0.0625 = 0.375$
- $\text{Gini}(\text{Temp}=\text{Mild}) = 1 - (4/6)^2 - (2/6)^2 = 1 - 0.444 - 0.111 = 0.445$
- We'll calculate weighted sum of gini index for temperature feature
- $\text{Gini}(\text{Temp}) = (4/14) \times 0.5 + (4/14) \times 0.375 + (6/14) \times 0.445 = 0.142 + 0.107 + 0.190 = 0.439$

Humidity

- Humidity is a binary class feature. It can be

Humidity	Yes	No	Number of instances
High	3	4	7
Normal	6	1	7

- $Gini(\text{Humidity}=\text{High}) = 1 - (3/7)^2 - (4/7)^2 = 1 - 0.183 - 0.326 = 0.489$
- $Gini(\text{Humidity}=\text{Normal}) = 1 - (6/7)^2 -$

Wind

- Wind is a binary class similar to humidity. It

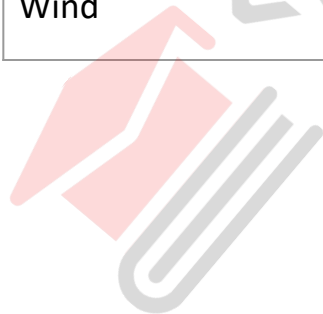
ca

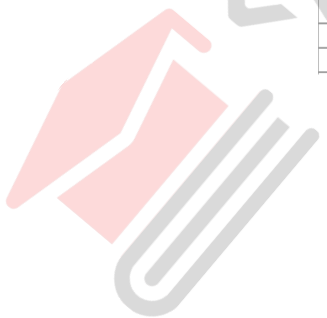
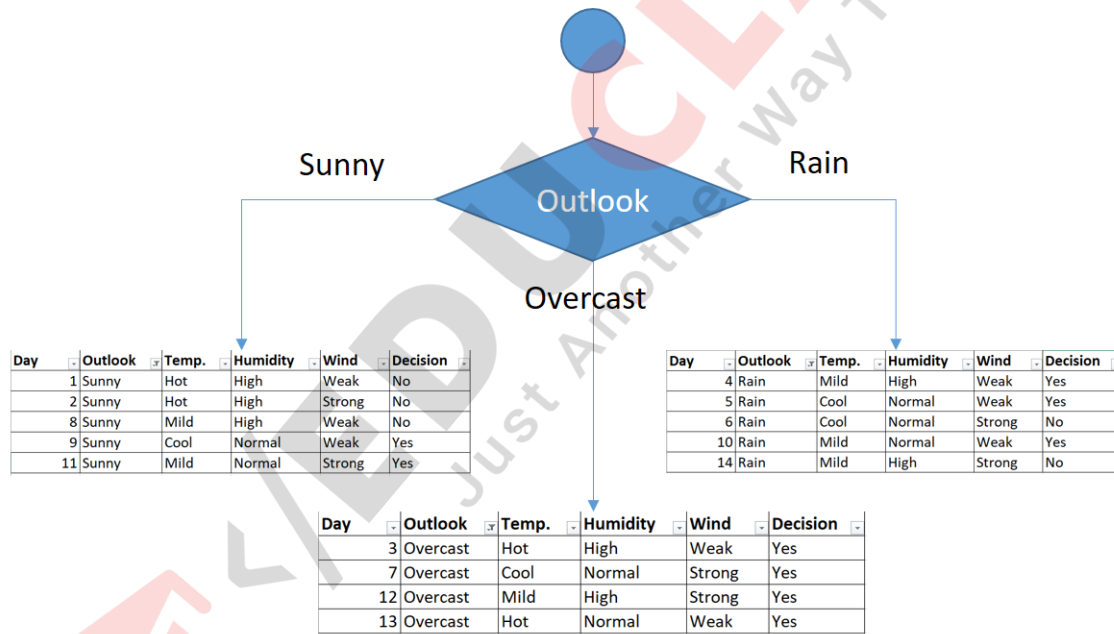
Wind	Yes	No	Number of instances
Weak	6	2	8
Strong	3	3	6

- $Gini(\text{Wind}=\text{Weak}) = 1 - (6/8)^2 - (2/8)^2 = 1 - 0.5625 - 0.0625 = 0.375$
- $Gini(\text{Wind}=\text{Strong}) = 1 - (3/6)^2 - (3/6)^2 = 1 -$

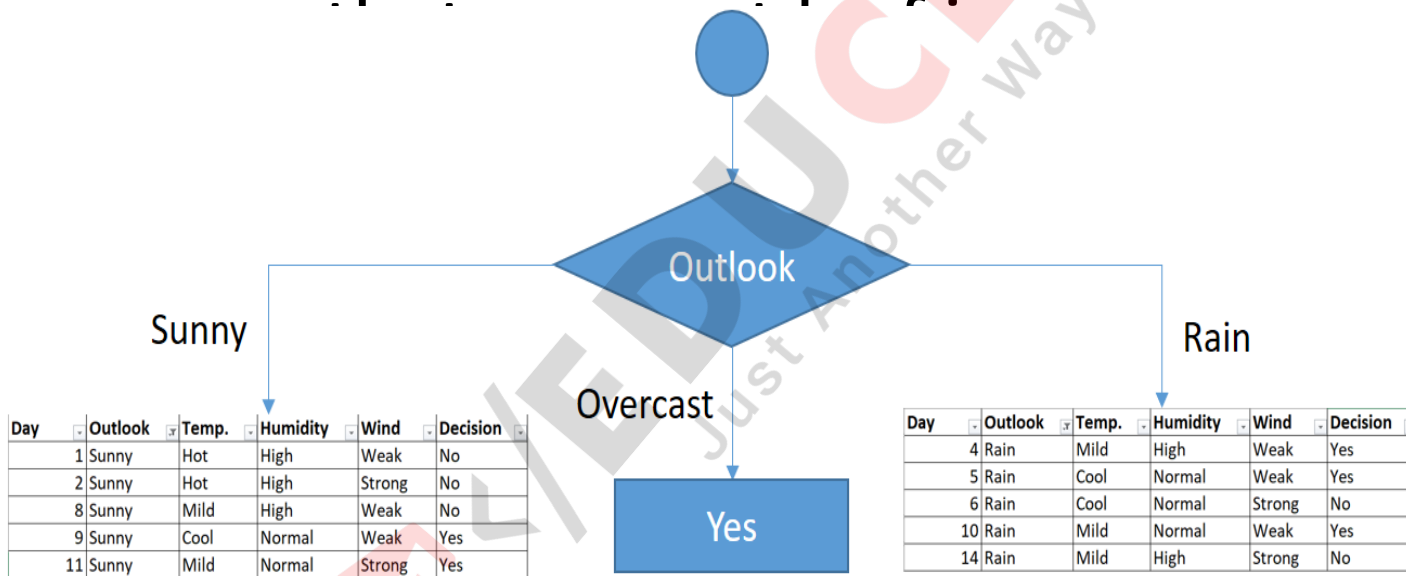
- We've calculated gini index values for each feature. The winner will be outlook feature because

Feature	Gini index
Outlook	0.342
Temperature	0.439
Humidity	0.367
Wind	0.428





- You might realize that sub dataset in the overcast leaf has only yes decisions. This



- Focus on the sub dataset for sunny outlook. We need to find the gini index scores for temperature, humidity and wind features respectively

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes



- **Gini of temperature for sunny outlook**

Temperature	Yes	No	Number of instances
Hot	0	2	2
Cool	1	0	1
Mild	1	1	2

- $\text{Gini}(\text{Outlook}=\text{Sunny and Temp.}=\text{Hot}) = 1 - (0/2)^2 - (2/2)^2 = 0$

- $\text{Gini}(\text{Outlook}=\text{Sunny and Temp.}=\text{Cool}) = 1 -$

- **Gini of humidity for sunny outlook**

Humidity	Yes	No	Number of instances
High	0	3	3
Normal	2	0	2

- $\text{Gini}(\text{Outlook}=\text{Sunny and Humidity}=\text{High}) = 1 - (0/3)^2 - (3/3)^2 = 0$
- $\text{Gini}(\text{Outlook}=\text{Sunny and Humidity}=\text{Normal}) = 1 - (2/2)^2 - (0/2)^2 = 0$

- **Gini of wind for sunny outlook**

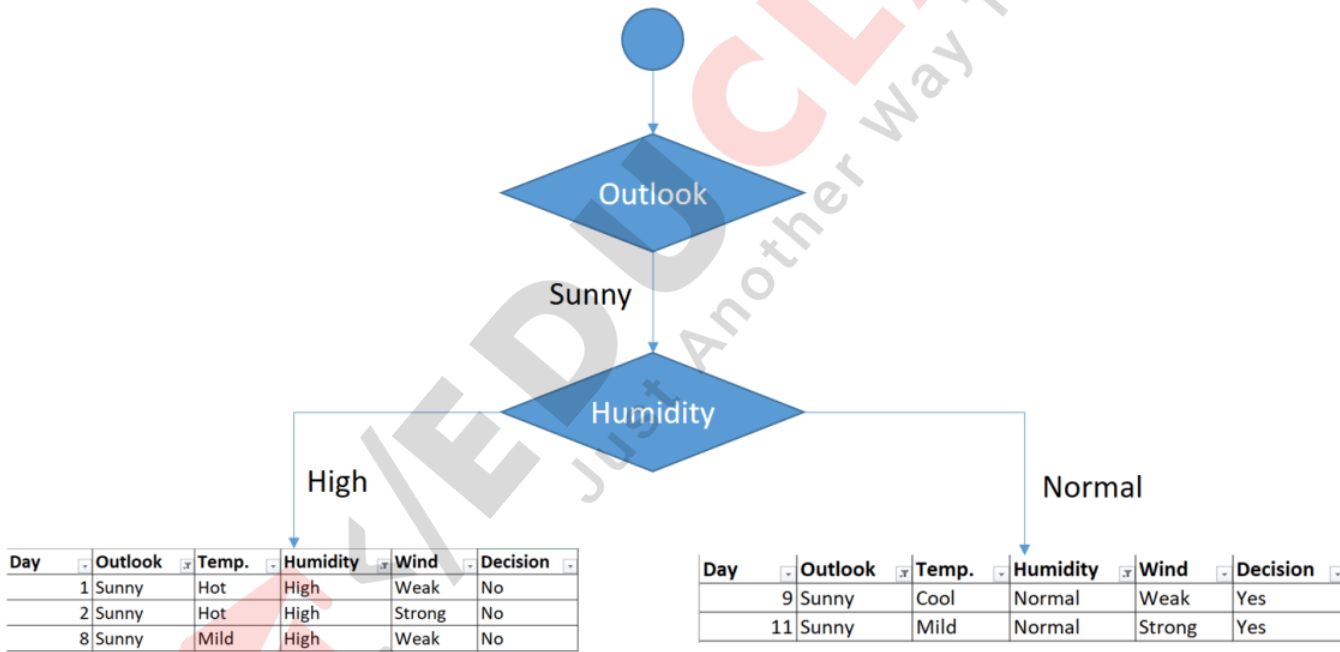
Wind	Yes	No	Number of instances
Weak	1	2	3
Strong	1	1	2

- $\text{Gini}(\text{Outlook}=\text{Sunny and Wind}=\text{Weak}) = 1 - (1/3)^2 - (2/3)^2 = 0.266$
- $\text{Gini}(\text{Outlook}=\text{Sunny and Wind}=\text{Strong}) = 1 - (1/2)^2 - (1/2)^2 = 0.2$
- $\text{Gini}(\text{Outlook}=\text{Sunny and Wind}) = (3/5) \times 0.266 + (2/5) \times 0.2 = 0.466$

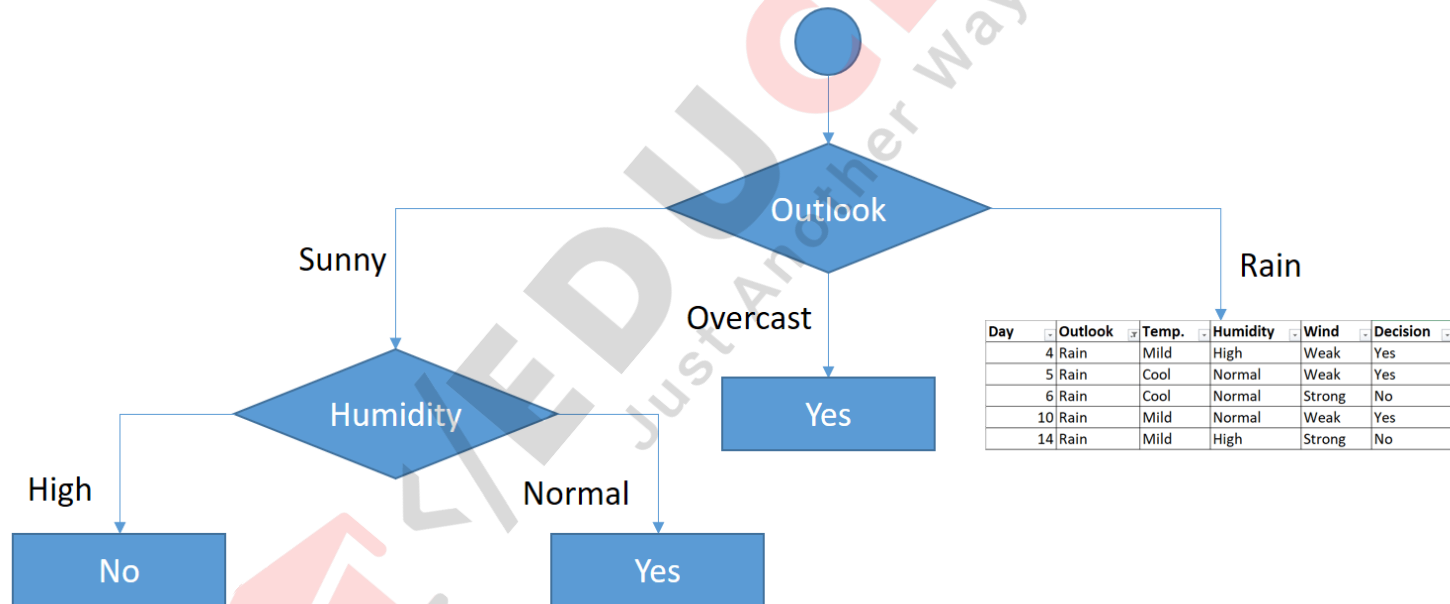
- **Decision for sunny outlook**
- We've calculated gini index scores for feature when outlook is sunny. The winner is humidity because it has the lowest value.

Feature	Gini index
Temperature	0.2
Humidity	0
Wind	0.466





- As seen, decision is always no for high humidity and sunny outlook. On the other



Rain outlook

Day	Outlook	Temp.	Humidity	Wind	Decision
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
10	Rain	Mild	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No



- We'll calculate gini index scores for temperature, humidity and wind features when outlook is rain.

- **Gini**

Temperature	Yes	No	Number of instances
Cool	1	1	2
Mild	2	1	3



- **Gini of humidity for rain outlook**

Humidity	Yes	No	Number of instances
High	1	1	2
Normal	2	1	3

- $\text{Gini}(\text{Outlook}=\text{Rain and Humidity}=\text{High}) = 1 - (1/2)^2 - (1/2)^2 = 0.5$
- $\text{Gini}(\text{Outlook}=\text{Rain and Humidity}=\text{Normal}) = 1 - (2/3)^2 - (1/3)^2 = 0.444$

FB/IG/TW: @educastudio • $\text{Gini}(\text{Outlook}=\text{Rain and Humidity}) =$

- **Gini of wind for rain outlook**

Wind	Yes	No	Number of instances
Weak	3	0	3
Strong	0	2	2

- $\text{Gini}(\text{Outlook}=\text{Rain and Wind}=\text{Weak}) = 1 - (3/3)^2 - (0/3)^2 = 0$

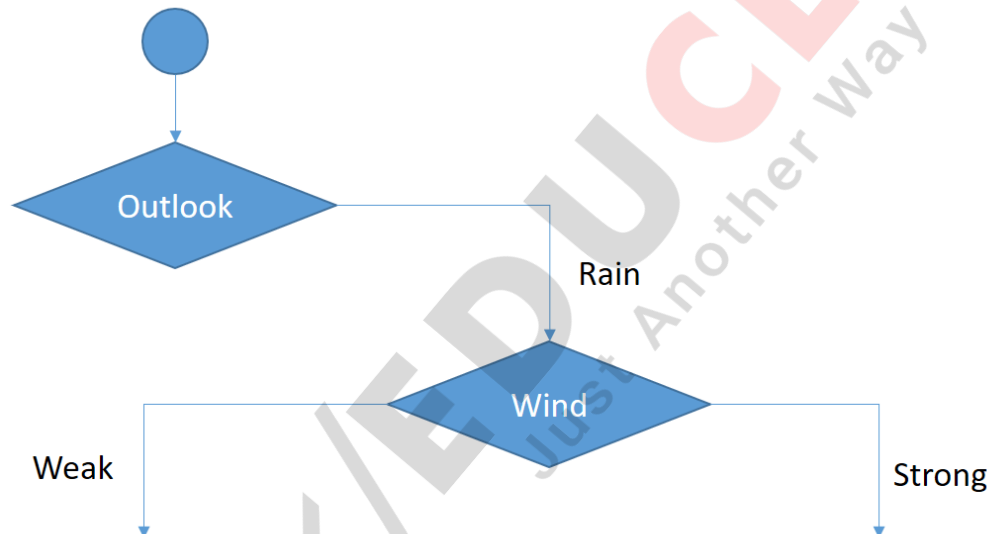
- $\text{Gini}(\text{Outlook}=\text{Rain and Wind}=\text{Strong}) = 1 - (0/2)^2 - (2/2)^2 = 0$

- The winner is wind feature for rain outlook because it has the minimum gini index score in features.

Feature	Gini index
Temperature	0.466
Humidity	0.466
Wind	0



- Put the wind feature for rain outlook branch and reiterate the new split datasets



Day	Outlook	Temp.	Humidity	Wind	Decision
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes

Day	Outlook	Temp.	Humidity	Wind	Decision
6	Rain	Cool	Normal	Strong	No
14	Rain	Mild	High	Strong	No

- As seen, decision is always yes when wind is weak. On the other hand, decision is always

