

Lecture 10

Machine Learning for IoT Data (I)

IS4151/IS5451 – AIoT Solutions and Development
AY 2024/25 Semester 2

Lecturer: A/P TAN Wee Kek

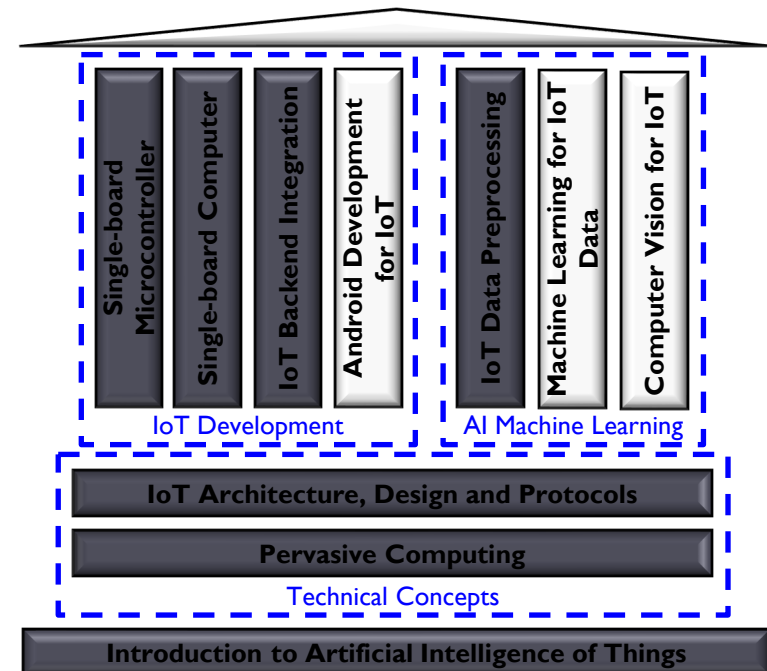
Email: tanwk@comp.nus.edu.sg :: **Tel:** 6516 6731 :: **Office:** COM3-02-35

Consultation: Tuesday, 2 pm to 4 pm. Additional consultations by appointment are welcome.



Quick Recap...

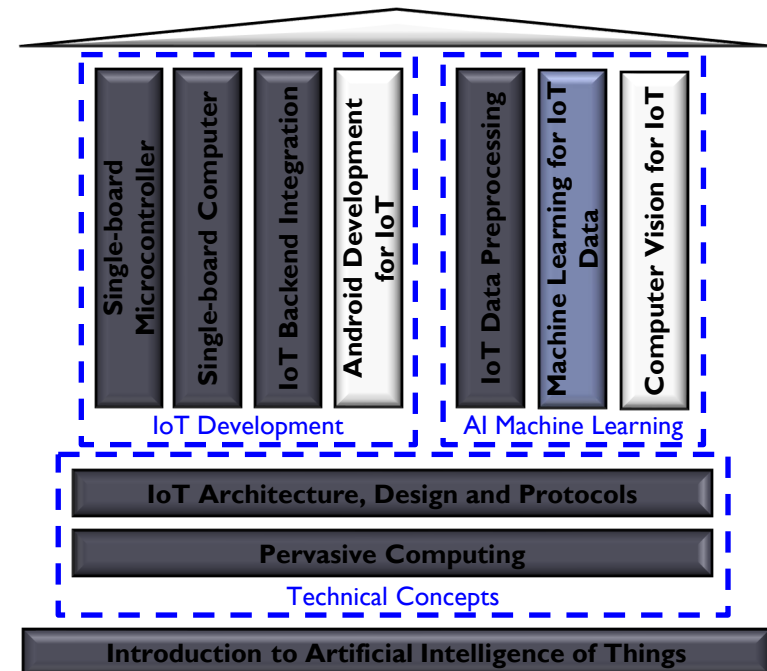
- ▶ In the previous lecture, we learnt:
 - ▶ More about supervised learning and unsupervised learning.
 - ▶ How to perform data preparation with Pandas.
 - ▶ How to perform data visualisation with Matplotlib.
- ▶ We now have sufficient knowledge to build machine learning models to enable smart AIoT systems.





Learning Objectives

- ▶ At the end of this lecture, you should understand:
 - ▶ How to perform prediction with regression analysis.
 - ▶ How to perform prediction with regression that involves categorical independent variables.
 - ▶ How to perform prediction with classification.





Readings

- ▶ Required readings:
 - ▶ None.
- ▶ Suggested readings:
 - ▶ None.

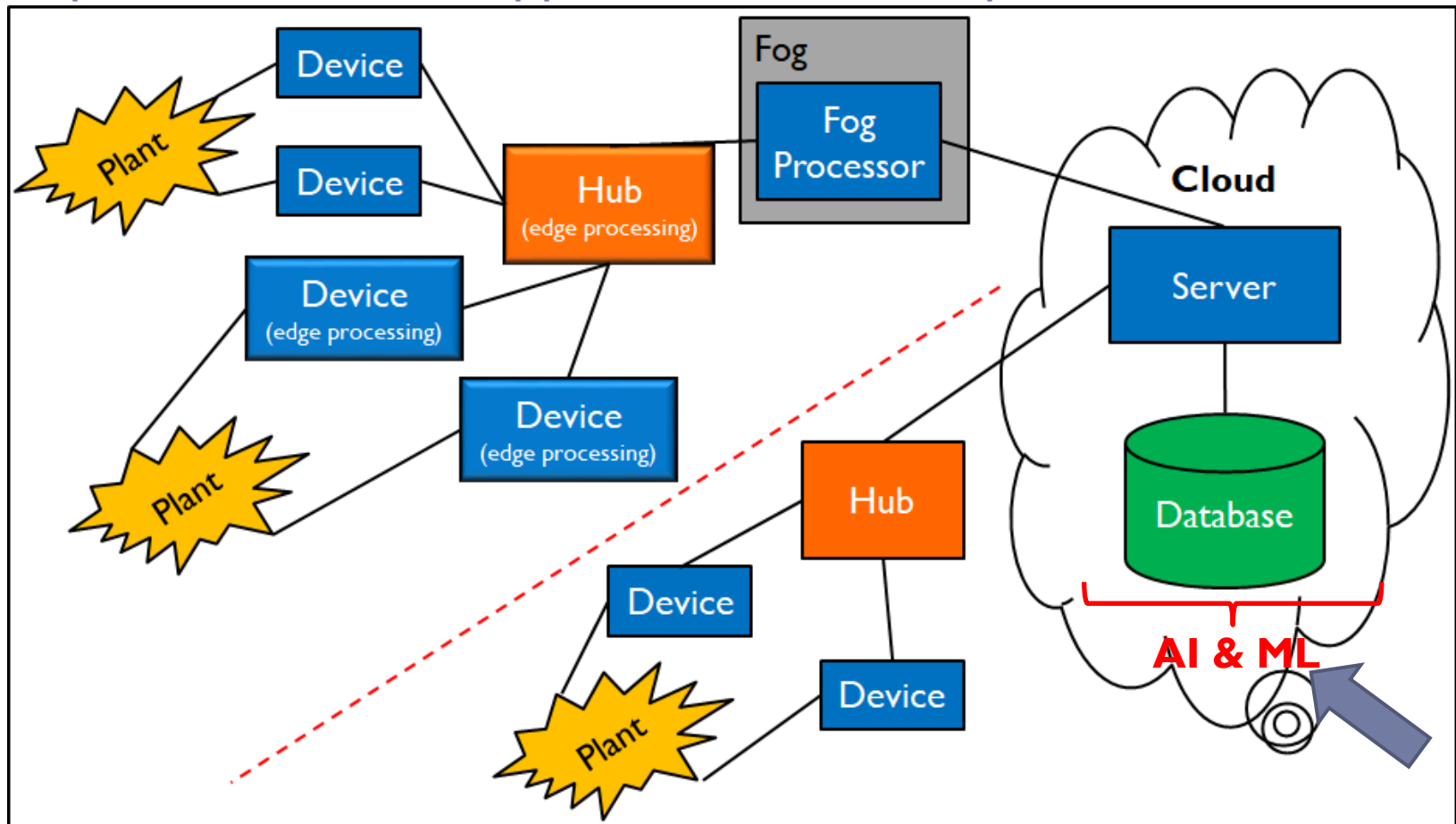


Technical Roadmap for IS4151/IS5451

Single-board Microcontroller
Android Wear

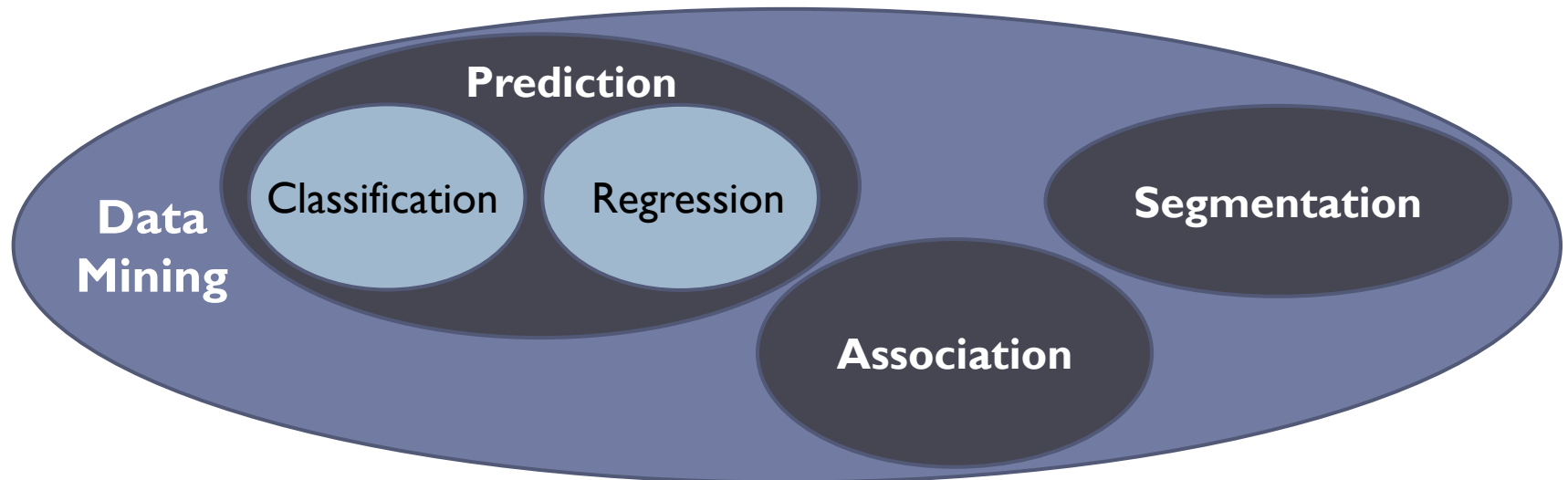
Single-board Computer
Android App

IoT Backend Integration



Recap on Machine Learning

- ▶ Recall that machine learning employs statistical and mathematical techniques to build a model based on sample data
- ▶ The objective is to identify patterns among variables in the data, i.e., data mining:
 - ▶ Data mining involves three main patterns:



Application of Regression to IoT Data

S/N	Application	Scenario	Regression Input/Output
1	Energy Consumption Forecasting	Smart meters in buildings or homes	Predict future energy usage (kWh) based on past consumption, weather data, and occupancy levels.
2	Predictive Maintenance	IoT sensors on machinery/equipment in factories.	Estimate Remaining Useful Life (RUL) or time until a part fails.
3	Environmental Monitoring	Sensors tracking temperature, humidity, air quality, etc.	Predict future environmental conditions (e.g., air pollution levels such as PM2.5 at a given time).
4	Smart Agriculture	IoT devices monitor soil moisture, temperature, light levels.	Predict crop yield (tons per hectare) or optimal irrigation volume.
5	Traffic Flow Prediction	Smart city sensors track vehicle counts, speeds, and flow rates.	Predict number of vehicles passing a junction or average speed in the next hour.

Application of Classification to IoT Data

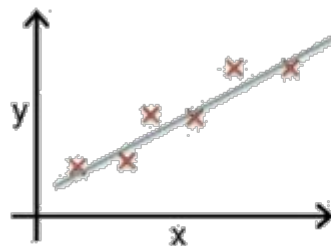
S/N	Application	Scenario	Classification Input/Output
1	Predictive Maintenance	Fault detection with sensors on industrial machines that monitor vibration, temperature, etc.	Determine if a machine is in a normal or faulty state.
2	Health Monitoring	Disease detection with wearable devices that collect data like heart rate, blood oxygen, and sleep patterns.	Identify risk of a health condition (e.g., arrhythmia, sleep apnea) – Healthy, At Risk, Needs Immediate Attention.
3	Smart City	Traffic violation detection with cameras and IoT sensors that track vehicles.	Automatically detect traffic violations (Speeding, Illegal Parking, Red Light Violation) or categorize vehicle types.
4	Smart Agriculture	Plant disease detection with sensors and cameras that monitor crops.	Detect whether a plant is <i>healthy</i> or affected by a specific disease (Healthy, Bacterial Blight, Leaf Spot, Rust).
5	Water Quality Monitoring	IoT sensors that monitor turbidity, pH, etc.	Classify water as safe or unsafe for drinking.

Prediction with Regression Analysis

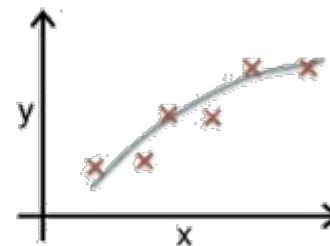
What is Regression Analysis?

- ▶ Builds statistical models that characterize relationships among (continuous) numerical variables:
 - ▶ Dependent variable must be numerical.
 - ▶ Non-numerical independent variables need to be converted into numerical variables.
- ▶ A regression model identifies a functional relationship between the dependent variable and independent variables:

$$Y = f(X_1, X_2, \dots, X_n)$$



Linear regression



Nonlinear regression

Linear Regression Model

- ▶ If we assume that the functional relationship is linear, we have linear regression models:
 - ▶ Most nonlinear relationship may be reduced to a linear one by a suitable transformation.
 - ▶ E.g., an exponential relationship $Y = e^{b+wX}$ can be linearized through a logarithmic transformation $Z = \log(Y)$ into a linear relationship $Z = b + wX$.
- ▶ A simple linear relationship has one independent variable and is of the form:

$$Y = \alpha + \beta X + \varepsilon$$

ε is a random variable known as error, which indicates the discrepancy between the response Y and the prediction

$$f(X) = \alpha + \beta X$$

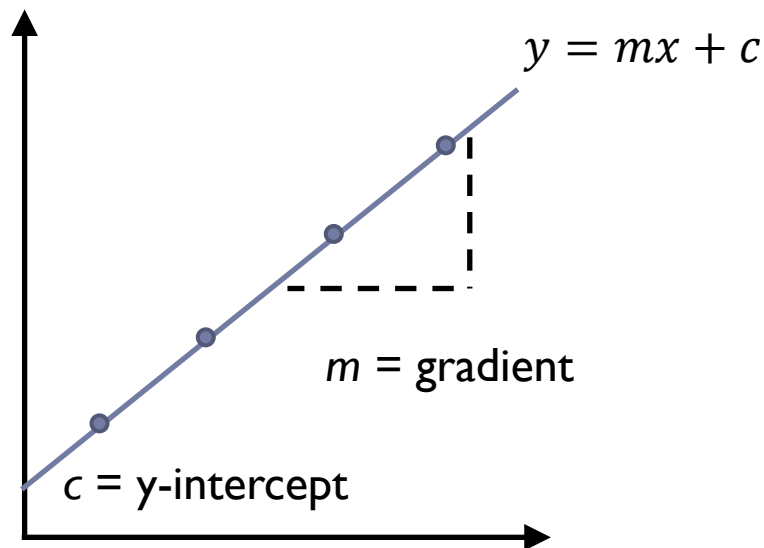
Linear Regression Model (cont.)

- ▶ A multiple linear relationship has multiple independent variable and is of the form:

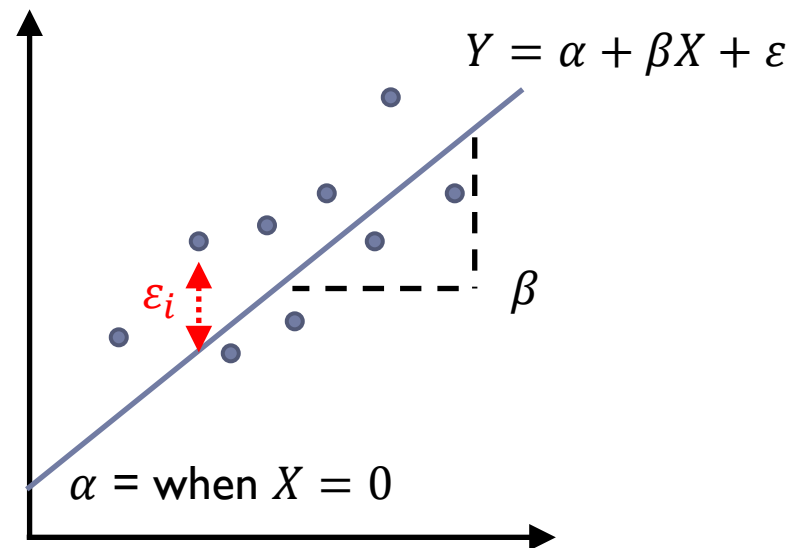
$$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n + \varepsilon$$

Simple Linear Regression

- ▶ Simple linear regression is analogue to linear equation:
 - ▶ In a linear equation $y = mx + c$, the points (x_n, y_n) lies exactly on a physical line.
 - ▶ In a simple linear regression $Y = \alpha + \beta X + \varepsilon$, we try to fit an imaginary line through the observations as best as possible.



Linear Equation



Simple Linear Regression

Simple Linear Regression (cont.)

- ▶ In simple linear regression, we want to minimize the error of the prediction:

$$\varepsilon_i = y_i - f(x_i) = y_i - \alpha - \beta x_i = y_i - \hat{y}_i$$

- ▶ The regression coefficients α and β can be computed by the method of least squares which minimizes the sum of the squared errors $SSE = \sum_{i=1}^S \varepsilon_i^2$
- ▶ This technique is known as the ordinary least squares (OLS) regression.

Example of Simple Linear Regression

- ▶ Predict a child's weight based on height:
 - ▶ The dataset contains 19 observations.
 - ▶ There are four variables altogether – Name, Weight (pound), Height (cm) and Age.
- ▶ In Python, we use both Scikit Learn and StatsModels to perform linear regression:
 - ▶ StatsModels provide more summary statistics as compared to Scikit Learn.
 - ▶ We could also manually calculate the required statistics...
- ▶ Refer to sample source file [src01](#) for the example.



Example of Simple Linear Regression (cont.)

	Weight	Height	Age
Name			
Alfred	69.0	112.5	14
Alice	56.5	84.0	13
Barbara	65.3	98.0	13
Carol	62.8	102.5	14
Henry	63.5	102.5	14
James	57.3	83.0	12
Jane	59.8	84.5	12
Janet	62.5	112.5	15
Jeffrey	62.5	84.0	13
John	59.0	99.5	12
Joyce	51.3	50.5	11
Judy	64.3	90.0	14
Louise	56.3	77.0	12
Mary	66.5	112.0	15
Philip	72.0	150.0	16
Robert	64.8	128.0	12
Ronald	67.0	133.0	15
Thomas	57.5	85.0	11
William	66.5	112.0	15

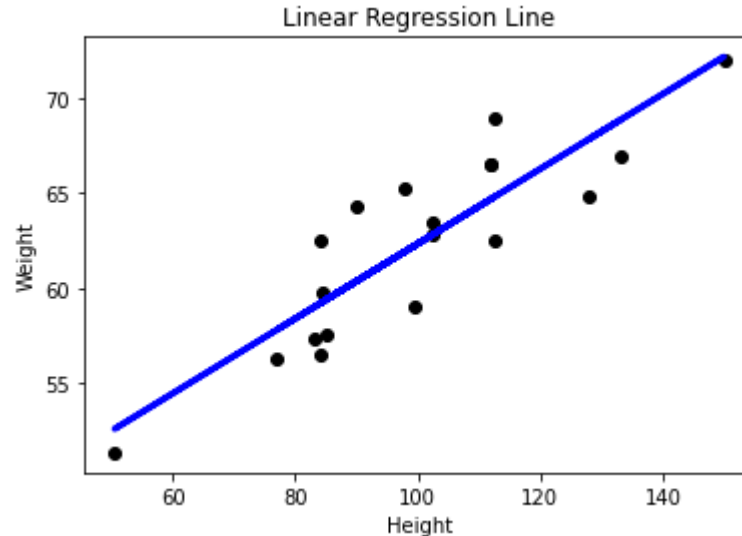
	Weight	Height	Age
Weight	1.000000	0.877785	0.811434
Height	0.877785	1.000000	0.740885
Age	0.811434	0.740885	1.000000

OLS Regression Results

Dep. Variable:	Weight	R-squared:	0.771			
Model:	OLS	Adj. R-squared:	0.757			
Method:	Least Squares	F-statistic:	57.08			
Date:	Sun, 04 Jul 2021	Prob (F-statistic):	7.89e-07			
Time:	20:23:27	Log-Likelihood:	-43.519			
No. Observations:	19	AIC:	91.04			
Df Residuals:	17	BIC:	92.93			
Df Model:	1					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	42.5701	2.680	15.885	0.000	36.916	48.224
Height	0.1976	0.026	7.555	0.000	0.142	0.253
Omnibus:	3.056	Durbin-Watson:	2.643			
Prob(Omnibus):	0.217	Jarque-Bera (JB):	1.596			
Skew:	0.408	Prob(JB):	0.450			
Kurtosis:	1.837	Cond. No.	474.			

Left to Right – Dataset, correlation matrix and regression results

Example of Simple Linear Regression (cont.)



Interpreting the regression model:

- The regression equation is $y = 42.5701 + 0.1976x + \varepsilon$
- $\beta = 0.1976$ implies a one unit increase in height leads to an expected increase of 0.1976 unit in weight.
- $\alpha = 42.5701$ implies that when $x = 0$, $y = 42.5701$ (danger of extrapolation).
- $N = 19$ is the number of observations – Most of the dots, i.e., actual (x_i, y_i) values, are close to the fitted line.

Example of Simple Linear Regression (cont.)

- ▶ Is the regression model good?
 - ▶ Analysis of Variance:
 - ▶ $F\text{-value} = 57.08$
 - ▶ The corresponding $p\text{-value}$ is < 0.0001 , indicating that at least one of the independent variables is useful for predicting the dependent variable.
 - ▶ In this case, since there is only one independent variable, i.e., the value of height is useful for predicting the value of weight.
 - ▶ Regression coefficient:
 - ▶ The corresponding $p\text{-value}$ for the intercept and regression coefficient is < 0.0001 .
 - ▶ We can reject the null hypotheses that the intercept and regression coefficient are zero.

Example of Simple Linear Regression (cont.)

- ▶ R-Square:

- ▶ Indicates the proportion of total variance explained by the independent variable.

- ▶ $R^2 = 0.771$

- ▶ Root Mean Squared Error (RMSE):

- ▶ Recall that in linear regression, the goal is to minimize SSE .

- ▶ $RMSE = \sqrt{\frac{SSE}{s}}$

- ▶ Thus, a smaller value of $RMSE$, i.e., close to 0.0, is better and indicates a model with better fit.

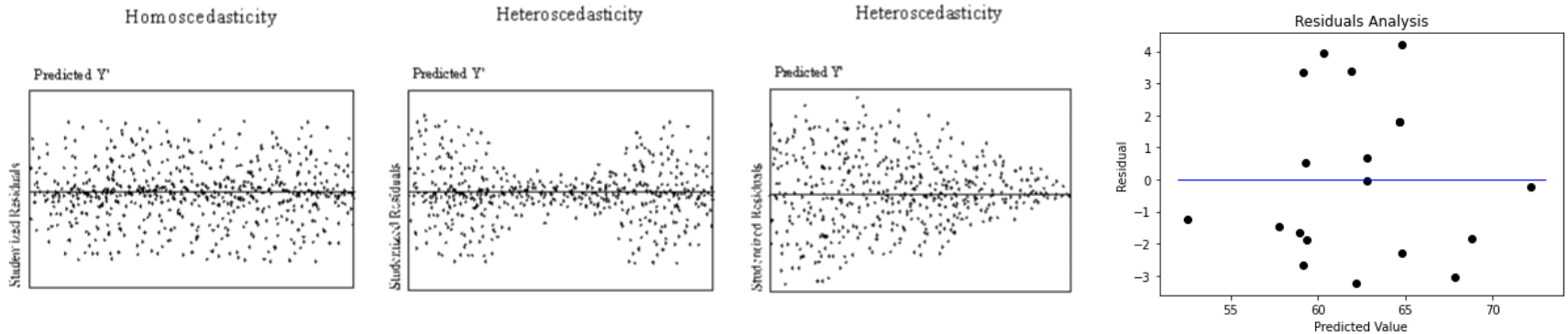
- ▶ $RMSE = 2.391$

- ▶ If weight is in Pound, the $RMSE$ would be 2.391 Pound (≈ 1.085 Kg).

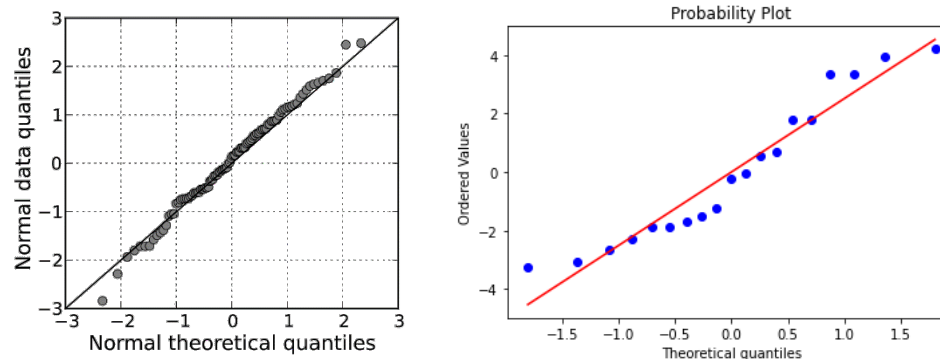
Validating the Assumptions of Linear Regression

- ▶ Linear regression has five key assumptions.
- ▶ Linear relationship:
 - ▶ Relationship between the independent and dependent variables is linear – Check regression line for linearity.
- ▶ Homoscedasticity:
 - ▶ Residuals are equal across the regression line.
 - ▶ Scatter plots between residuals and predicted values are used to confirm this assumption.
 - ▶ Any pattern would result in a violation of this assumption and point toward a poor fitting model.
 - ▶ Refer to sample source file [src02](#) for the example.

Validating the Assumptions of Linear Regression (cont.)



- ▶ We can also check the normality of the residuals using a Q-Q plot – Data points must fall (approximately) on a straight line for normal distribution.
- ▶ Refer to sample source file [src03](#) for the example.

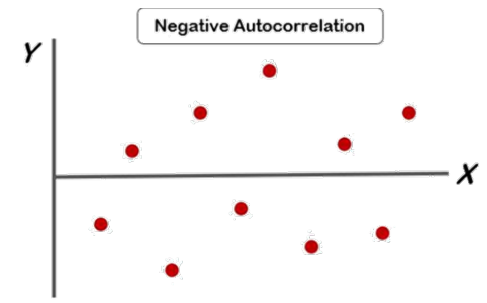
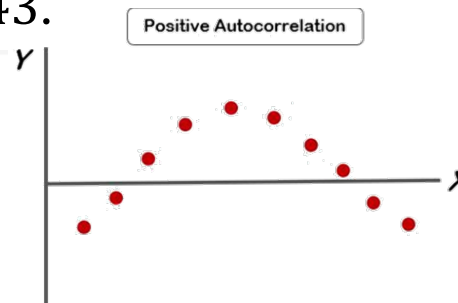


Validating the Assumptions of Linear Regression (cont.)

▶ Auto-correlation

- ▶ Residuals must be independent from each other.
- ▶ Scatter plots between residuals and predicted values – Residuals are randomly distributed with no pattern.
- ▶ We can also use the Durbin-Watson test to test the null hypothesis that the residuals are not linearly auto-correlated:
 - ▶ The test statistic lies in the range $0 \leq d \leq 4$ and $d \approx 2$ indicates no autocorrelation.
 - ▶ $d < 2$ and $d > 2$ indicates positive and negative autocorrelation
 - ▶ In our example, $d = 2.643$.

Omnibus: 3.056	Durbin-Watson: 2.643
Prob(Omnibus): 0.217	Jarque-Bera (JB): 1.596
Skew: 0.408	Prob(JB): 0.450
Kurtosis: 1.837	Cond. No. 474.



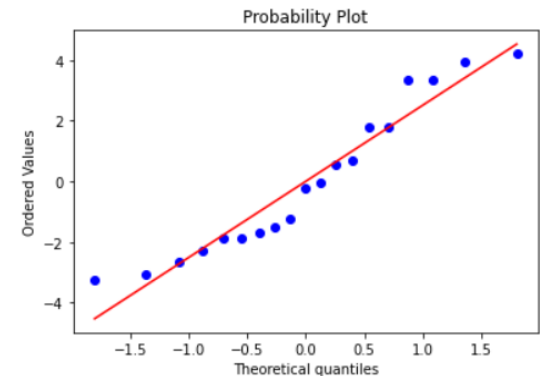
Validating the Assumptions of Linear Regression (cont.)

► Multivariate normality:

- Residuals must be normally distributed.
- We can perform visual/graphical test to check for normality of the data using Q-Q plot and also histogram.
- We can also validate statistically using the Kolmogorov–Smirnov test.
- Refer to sample source file [src04](#) (similar to [src03](#)) for the example.

```
stats.probplot(residuals, dist="norm", plot=plt)
plt.show()
```

✓ 0.1s



```
a,b = stats.kstest(residuals, 'norm')
print('Statistic = {}, p-value = {}'.format(a, b))
```

✓ 0.0s

Statistic = 0.3679795429619702, p-value = 0.008157223308320027

Validating the Assumptions of Linear Regression (cont.)

- ▶ **Multicollinearity:**

- ▶ Independent variables are not correlated with each other.
- ▶ For simple linear regression, this is not an issue.

- ▶ **Child's weight example:**

- ▶ We may conclude that the residuals are independent but not normally distributed.
- ▶ The linear regression model generally fits the data well.

Multiple Linear Regression

- ▶ Limitations of simple linear regression:

- ▶ In many real-world scenarios, there are likely more than one independent variables that are correlated with the dependent variable.
- ▶ Multiple linear regression allows us to handle such scenarios.

- ▶ Recall that multiple linear regression is of the form:

$$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n + \varepsilon$$

- ▶ The regression coefficient β_j expresses the marginal effect of the variable X_j on the dependent variable Y , conditioned on the current value of the remaining independent variables.
- ▶ Scale of the values influences the value of the corresponding regression coefficient and thus it might be useful to standardize the independent variables.

Example of Multiple Linear Regression

- ▶ Predict colleges and universities graduation rate:
 - ▶ Predict the percentage of students accepted into a college program who would eventually graduate.
 - ▶ The dataset contains 49 observations.
 - ▶ There are five possible independent variables altogether:
 - ▶ Type – Type of college, i.e., University or Liberal Arts.
 - ▶ MedianSAT
 - ▶ AcceptanceRate
 - ▶ ExpendituresPerStudent
 - ▶ Top10PercentHS – Proportion of accepted students who are among the top 10% of their high school cohort.
 - ▶ We will exclude Type as it is a nominal categorical variable.
- ▶ Refer to sample source file [src05](#) for the example.

Example of Multiple Linear Regression (cont.)

	Median SAT	AcceptanceRate	ExpendituresPerStudent	Top10PercentHS	GraduationPercent
Median SAT	1.000000	-0.601902	0.572742	0.503468	0.564147
AcceptanceRate	-0.601902	1.000000	-0.284254	-0.609721	-0.550378
ExpendituresPerStudent	0.572742	-0.284254	1.000000	0.505782	0.042504
Top10PercentHS	0.503468	-0.609721	0.505782	1.000000	0.138613
GraduationPercent	0.564147	-0.550378	0.042504	0.138613	1.000000

Dep. Variable:	GraduationPercent	R-squared:	0.534
Model:	OLS	Adj. R-squared:	0.492
Method:	Least Squares	F-statistic:	12.63
Date:	Sun, 04 Jul 2021	Prob (F-statistic):	6.33e-07
Time:	21:49:14	Log-Likelihood:	-148.69
No. Observations:	49	AIC:	307.4
Df Residuals:	44	BIC:	316.8
Df Model:	4		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
const	17.9210	24.557	0.730	0.469	31.571	67.413
Median SAT	0.0720	0.018	4.004	0.000	0.036	0.108
AcceptanceRate	-24.8592	8.315	-2.990	0.005	41.617	-8.101
ExpendituresPerStudent	-0.0001	6.59e-05	-2.057	0.046	-0.000	-2.77e-06
Top10PercentHS	-0.1628	0.079	-2.051	0.046	-0.323	-0.003

Omnibus:	1.954	Durbin-Watson:	2.010
Prob(Omnibus):	0.376	Jarque-Bera (JB):	1.833
Skew:	-0.450	Prob(JB):	0.400
Kurtosis:	2.706	Cond. No.	1.12e+06

Top to Bottom –
Correlation matrix and
regression results

Example of Multiple Linear Regression (cont.)

- ▶ Interpreting the regression model:

- ▶ The regression equation is:

GraduationPercent

$$\begin{aligned} &= 17.921 + 0.0720\textit{MedianSAT} - 24.8592\textit{AcceptanceRate} \\ &- 0.0001\textit{ExpendituresPerStudent} \\ &- 0.1628\textit{Top10PercentHS} \end{aligned}$$

- ▶ Interpreting the regression coefficients:

- ▶ Higher median SAT scores and lower acceptance rates suggest higher graduation rates.
 - ▶ 1 unit increase in median SAT scores increase graduation rates by 0.0720 unit, all other things being equal.
 - ▶ 1 unit increase in acceptance rates decrease graduation rates by 24.8592 unit, all other things being equal.

Example of Multiple Linear Regression (cont.)

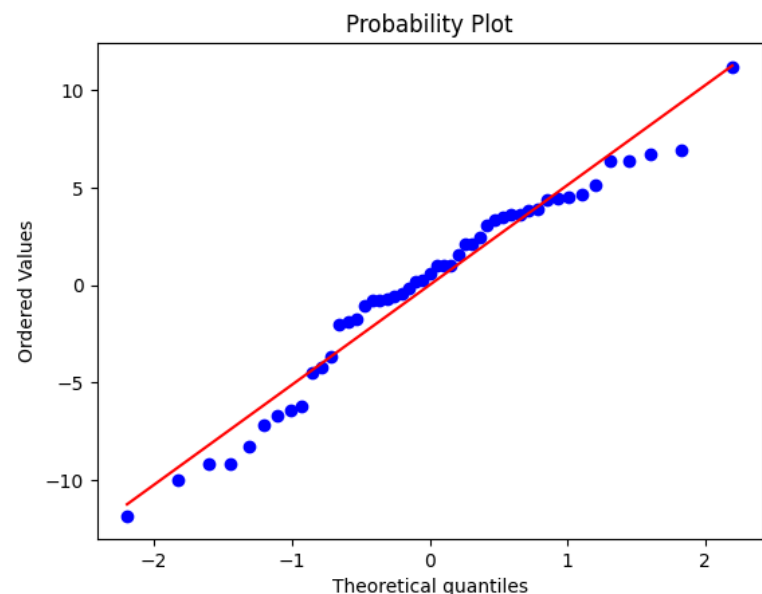
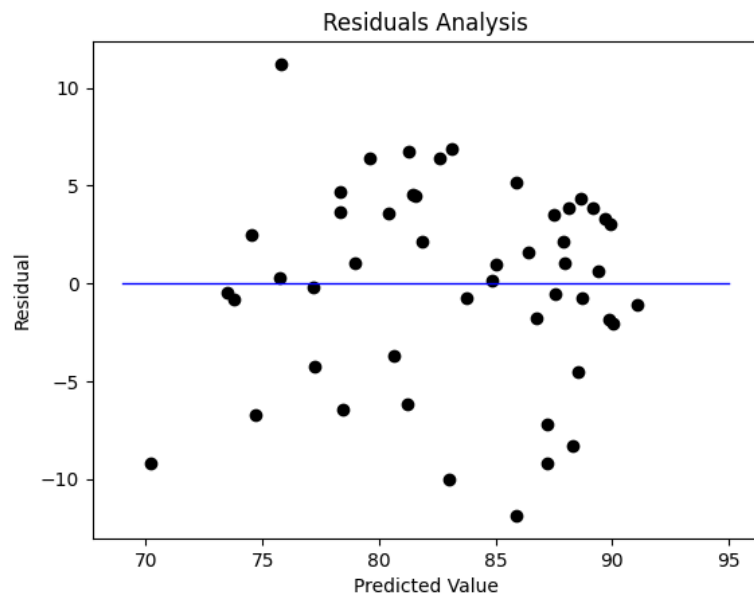
► Overall:

- $F\text{-value} = 12.63$ with a corresponding $p\text{-value} < 0.0001$
- The corresponding $p\text{-value}$ for the intercept is ≥ 0.05 but those for the regression coefficient is < 0.05
- Do the regression coefficients and their signs make sense?
- $R^2 = 0.534$
- R^2 and Adjusted R^2 are quite similar:
 - In a multiple linear regression model, this means that the inclusion of multiple independent variables is useful.
- $RMSE = 5.03$

Evaluating the Assumptions of Linear Regression

► Residuals analysis:

- The scatter plot of residuals against predicted values and the Q-Q plot of the residuals shows that the residuals are independent but not normally distributed.
- See the sample script [src06](#).



Evaluating the Assumptions of Linear Regression (cont.)

► Multicollinearity:

- Occurs when significant linear correlation exists between two or more predictive variables.
- Potential problems:
 - Regression coefficients are inaccurate.
 - Compromises overall significance of the model.
 - Possible that the coefficient of determination is close to 1 while the regression coefficients are not significantly different from 0.
- Pairwise linear correlation coefficients may be calculated.

	MedianSAT	AcceptanceRate	ExpendituresPerStudent	Top10PercentHS	GraduationPercent
MedianSAT	1.000000	-0.601902	0.572742	0.503468	0.564147
AcceptanceRate	-0.601902	1.000000	-0.284254	-0.609721	-0.550378
ExpendituresPerStudent	0.572742	-0.284254	1.000000	0.505782	0.042504
Top10PercentHS	0.503468	-0.609721	0.505782	1.000000	0.138613
GraduationPercent	0.564147	-0.550378	0.042504	0.138613	1.000000

Evaluating the Assumptions of Linear Regression (cont.)

- ▶ We can observe mild correlation among the four independent variables.
- ▶ Consequently, you would see that even though R^2 is moderately high, two of the independent variables, i.e., `ExpenditurePerStudent` and `Top10PercentHS`, are just below the 0.05 (or 95%) significance threshold.

Evaluating the Assumptions of Linear Regression (cont.)

- ▶ To identify multiple linear relationships or multicollinearity among predictive variables:
 - ▶ Calculate the **variance inflation factor (VIF)** for each predictor X_j as:

$$VIF_j = \frac{1}{1 - R_j^2}$$

where R_j^2 is the coefficient of determination for the model that explains X_j , treated as a response, through the remaining independent variables.

- ▶ $VIF_j > 5$ indicates multicollinearity.
- ▶ VIF can be calculated in StatsModels – See the sample script [src07](#).

	VIF Factor	features
0	81.815558	MedianSAT
1	10.431692	AcceptanceRate
2	6.444369	ExpendituresPerStudent
3	57.034010	Top10PercentHS

Revisiting the Child's Weight Prediction Case Study

- ▶ From the perspective of multiple linear regression, would Age be a useful predictor of Weight in addition to Height?
- ▶ Recall the correlation matrix:

	Weight	Height	Age
Weight	1.000000	0.877785	0.811434
Height	0.877785	1.000000	0.740885
Age	0.811434	0.740885	1.000000

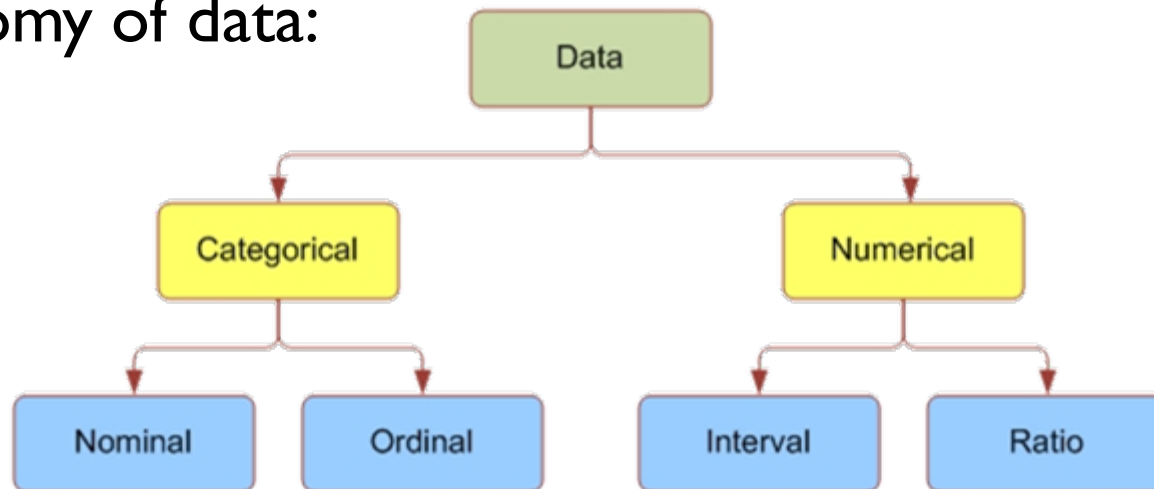
- ▶ Refer to sample source file [src08](#) for the example.



Encoding Categorical Independent Variables for Regression

Recap on Regression Analysis...

- ▶ Recall that regression analysis builds statistical models that characterize relationships among (continuous) numerical variables:
 - ▶ Dependent variable must be numerical.
 - ▶ Non-numerical independent variables can be converted into numerical variables.
- ▶ Taxonomy of data:



Categorical Data

▶ **Categorical Data:**

- ▶ Represent the labels of multiple classes used to divide a variable into specific groups.
- ▶ E.g., race, sex, age group and educational level.

▶ **Nominal data:**

- ▶ Categorical variables without natural ordering.
- ▶ E.g., Marital status can be categorized into (1) single, (2) married and (3) divorced.

▶ **Ordinal data:**

- ▶ Categorical variables that lend themselves to natural ordering.
 - ▶ More specifically, the codes assigned to objects represent the rank order among them.

Categorical Data (cont.)

- ▶ But it makes no sense to calculate the differences or ratios between values.
- ▶ E.g., credit score can be categorized as (1) low, (2) medium and (3) high.
- ▶ However, the additional rank-order information is useful in certain machine learning algorithms for building a better model.

Treatment of Categorical Independent Variables

- ▶ Categorical variables may be included as predictors in a regression model using dummy variables.
- ▶ A nominal categorical variable X_j with H distinct values denote by $V = \{v_1, v_2, \dots, v_H\}$ may be represented in two ways:
 - ▶ Using arbitrary numerical values:
 - ▶ Regression coefficients will be affected by the chosen scale.
 - ▶ Compromises significance of the model.
 - ▶ Using $H - 1$ binary variables $D_{j1}, D_{j2}, \dots, D_{j,H-1}$ known as dummy variables:
 - ▶ Each binary variable D_{jh} is associated with level v_h of X_j .
 - ▶ D_{jh} takes the value of 1 if $x_{ij} = v_h$.

Data Encoding Examples with Multiple Linear Regression

► Nominal categorical data:

- Housing estate – Multinomial variable with 5 different possible values represented using 4 binomial variables I_1 to I_4 :

Housing Estate	I_1	I_2	I_3	I_4
Ang Mo Kio	1	0	0	0
Bishan	0	1	0	0
Clementi	0	0	1	0
Dover	0	0	0	1
East Coast	0	0	0	0

- Predict price of HDB flats (y) based on the size (x_1), floor level (x_2) and distance to MRT station (x_3).
- Model: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 I_1 + \beta_5 I_2 + \beta_6 I_3 + \beta_7 I_4$
- In East Coast, the model is actually: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3$

Data Encoding Examples with Multiple Linear Regression (cont.)

- ▶ In Dover, the model is: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_7$
- ▶ Suppose $\beta_7 = -12000$: The price of a HDB flat in Dover is expected to be \$12000 less than another HDB flat in East Coast if both HDB flats have the same size, are on the same level and have the same distance to the nearest MRT station.
- ▶ Interpretation of the dummy variables always with respect to East Coast.



Data Encoding Examples with Multiple Linear Regression (cont.)

► Ordinal categorical data:

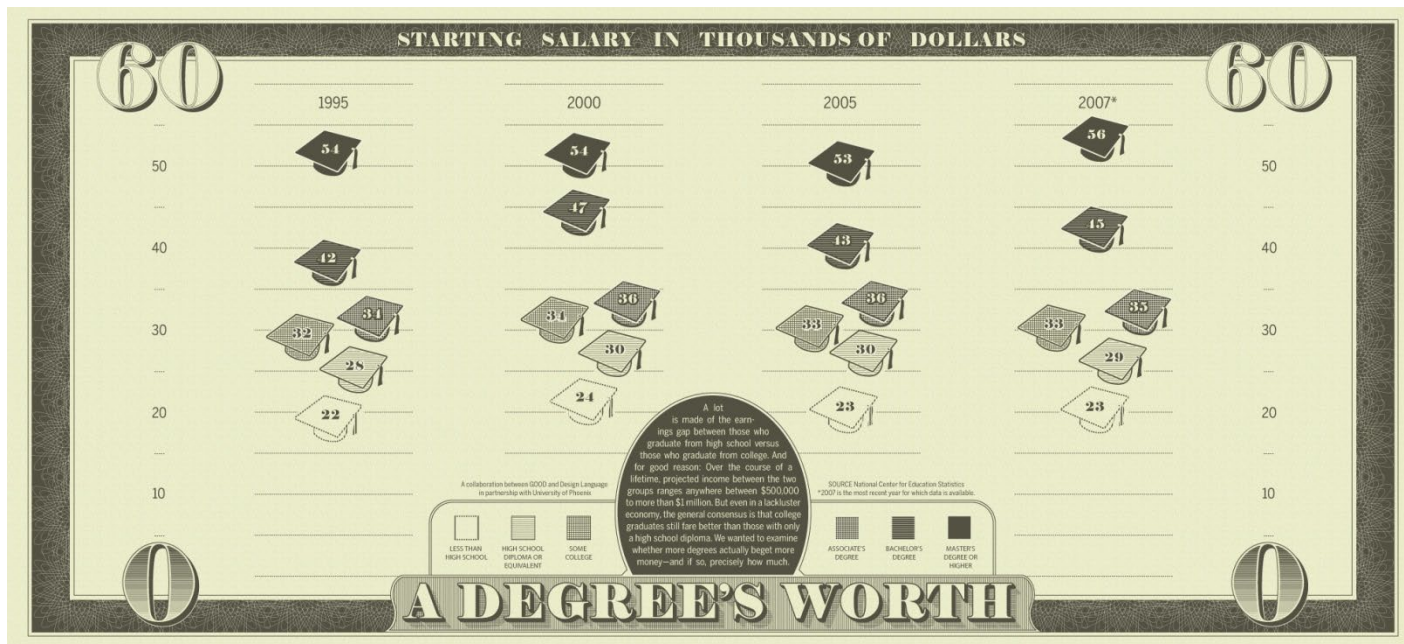
- Education level – Multinomial variable with 4 different possible values represented using 3 binomial variables I_1 to I_3 :

Education Level	I_1	I_2	I_3
Elementary School	0	0	0
High School	1	0	0
College	1	1	0
Graduate School	1	1	1

- Predict starting salary (y) based on education level.
- Model: $y = \beta_0 + \beta_1 I_1 + \beta_2 I_2 + \beta_3 I_3$
- Suppose $\beta_1 = \beta_2 = \beta_3 = 0$: No difference in starting salary.

Data Encoding Examples with Multiple Linear Regression (cont.)

- Suppose $\beta_1 = 0, \beta_2 > 0, \beta_3 = 0$:
 - Model: $y = \beta_0 + \beta_2 I_2$
 - When education level is high school or lower, $y = \beta_0$
 - When education level is college or higher, $y = \beta_0 + \beta_2$



Example of Categorical Independent Variables

- ▶ **Predict colleges and universities graduation rate:**
 - ▶ We will now include Type as one of the independent variables.
 - ▶ Type is a nominal categorical variable with two values.
 - ▶ Values of the variable Type are {Lib Arts, University}, i.e., $H = 2$
 - ▶ Thus, we use one dummy variable University with a numeric 1 representing University and a numeric 0 representing Lib Arts.
 - ▶ A University has a 1.4363 unit higher graduation rate compared to Lib Arts, all other things being equal.
 - ▶ But the regression coefficient of University is not significant in this case ($p = 0.524$).

Example of Categorical Independent Variables (cont.)

Dummy coding of “University”

src09

```
[ ]: import math

import pandas as pd

from sklearn.linear_model import LinearRegression
from sklearn import metrics

import statsmodels.api as sm

pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

[ ]: df = pd.read_csv('../data/colleges.csv')
df['University'] = 0

for i in range(0, len(df.index)):
    if df.iloc[i]['Type'] == 'University':
        df.loc[i, 'University'] = 1

df

[ ]: df = df.drop('School', axis=1)
df = df.drop('Type', axis=1)
df

[ ]: independent_variables = df.drop('GraduationPercent', axis=1)

x = independent_variables.values
y = df['GraduationPercent'].values

lr = LinearRegression(fit_intercept = True)
lr.fit(x, y)
y_pred = lr.predict(x)

print('Coefficients = ', lr.coef_)
```

Example of Categorical Independent Variables (cont.)

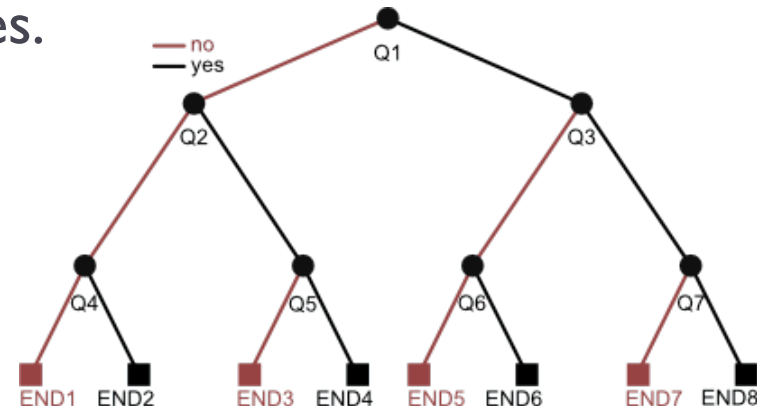
OLS Regression Results

Dep. Variable:	GraduationPercent	R-squared:	0.539			
Model:	OLS	Adj. R-squared:	0.485			
Method:	Least Squares	F-statistic:	10.05			
Date:	Sun, 25 Jul 2021	Prob (F-statistic):	2.02e-06			
Time:	21:12:30	Log-Likelihood:	-148.45			
No. Observations:	49	AIC:	308.9			
Df Residuals:	43	BIC:	320.3			
Df Model:	5					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	12.5912	26.078	0.483	0.632	-40.000	65.183
MedianSAT	0.0777	0.020	3.856	0.000	0.037	0.118
AcceptanceRate	-24.6380	8.378	-2.941	0.005	-41.534	-7.741
ExpendituresPerStudent	-0.0002	7.98e-05	-2.056	0.046	-0.000	-3.13e-06
Top10PercentHS	-0.1866	0.088	-2.119	0.040	-0.364	-0.009
University	1.4363	2.236	0.642	0.524	-3.073	5.946
Omnibus:	1.261	Durbin-Watson:	2.001			
Prob(Omnibus):	0.532	Jarque-Bera (JB):	1.113			
Skew:	-0.358	Prob(JB):	0.573			
Kurtosis:	2.817	Cond. No.	1.18e+06			

Prediction with Classification

Decision Trees

- ▶ The best known and most widely used learning methods in data mining applications.
- ▶ Reasons for its popularity include:
 - ▶ Conceptual simplicity.
 - ▶ Ease of usage.
 - ▶ Computational speed.
 - ▶ Robustness with respect to missing data and outliers.
 - ▶ Interpretability of the generated rules.

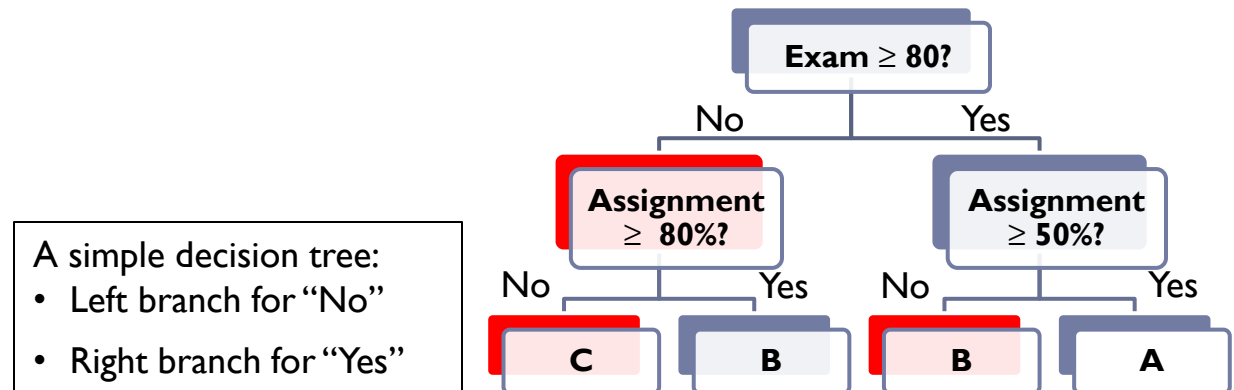


Decision Trees (cont.)

- ▶ The **development of a decision tree** involves recursive, heuristic, top-down induction:
 1. Initialization phase – All observations are placed in the root of the tree. The root is placed in the active node list L .
 2. If the list L is empty, stop the procedure. Otherwise, node $J \in L$ is selected, removed from the list and used as the node for analysis.
 3. The optimal rule to split the observations in J is then determined, based on an appropriate preset criterion:
 - ▶ If J does not need to be split, node J becomes a leaf, target class is assigned according to majority class of observations.
 - ▶ Otherwise, split node J , its children are added to the list.
 - ▶ Go to Step 2.

Components of Decision Trees

- ▶ Components of the top-down induction of decision trees:
 - ▶ **Splitting rules** – Optimal way to split a node (i.e., assigning observations to child nodes) and for creating child nodes.
 - ▶ **Stopping criteria** – If the node should be split or not. If not, this node becomes a leaf of the tree.
 - ▶ **Pruning criteria** – Avoid excessive growth of the tree during tree generation phase (pre-pruning) and reduce the number of nodes after the tree has been generated (post-pruning).



Example of a Decision Tree


- ▶ Given the dataset:

Observation #	Income	Credit Rating	Loan Risk
0	23	High	High
1	17	Low	High
2	43	Low	High
3	68	High	Low
4	32	Moderate	Low
5	20	High	High

- ▶ The task is to predict Loan-Risk.
- ▶ We will be using the univariate binary splitting approach.

Example of a Decision Tree (cont.)

- ▶ Given the data set D , we start building the tree by creating a root node.
- ▶ If this node is sufficiently “pure”, then we stop.
- ▶ If we do stop building the tree at this step, we use the majority class to classify/predict.
- ▶ In this example, we classify all patterns as having Loan-Risk = “High”.
- ▶ Correctly classify 4 out of 6 input samples to achieve classification accuracy of: $(4/6) \times 100\% = 66.67\%$
- ▶ This node is split according to impurity measures:
 - ▶ Gini Index (used by [CART](#))
 - ▶ Entropy (used by [ID3](#), [C4.5](#), [C5](#))



Loan-Risk = High
Acc = 66.67%

Using Gini Index

- ▶ CART (Classification and Regression Trees) uses the **Gini index** to measure the impurity of a dataset:
 - ▶ Gini index for the observations in node q is:

$$Gini(q) = 1 - \sum_{h=1}^H p_h^2$$

where

q is the node that contains Q examples from H classes

p_h is a relative frequency of class h in node q

- ▶ In our dataset, there are 2 classes High and Low, $H = 2$.

$$p_{High} = \frac{4}{4+2} = \frac{2}{3} \quad p_{Low} = \frac{2}{4+2} = \frac{1}{3}$$


$$Gini(q) = 1 - \left(\frac{2}{3} \times \frac{2}{3} \right) - \left(\frac{1}{3} \times \frac{1}{3} \right) = \frac{4}{9} = 0.4444$$



Using Gini Index (cont.)

- ▶ Should Income be used as the variable to split the root node?
- ▶ Income is a variable with continuous values.
- ▶ Sort the data according to Income values:

	Observation #	Income	Credit Rating	Loan Risk
	1	17	Low	High
	5	20	High	High
Split 1	0	23	High	High
Split 2	4	32	Moderate	Low
Split 3	2	43	Low	High
	3	68	High	Low



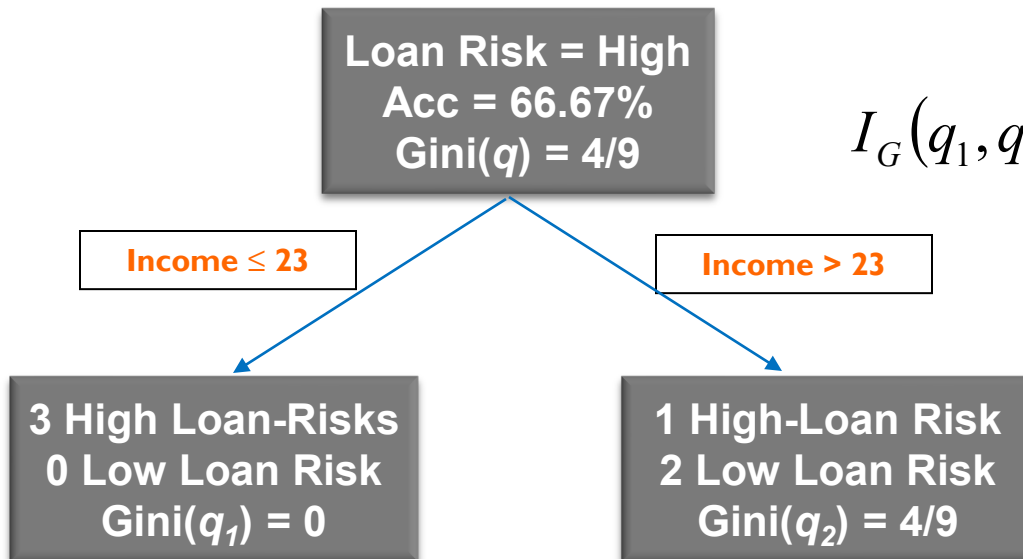
Using Gini Index (cont.)

- ▶ We consider 3 possible splits when there are changes in the value of Loan-Risk.
 - ▶ Case I – Split condition $\text{Income} \leq 23$ versus $\text{Income} > 23$

Impurity after split:

$$I(q_1, q_2, \dots, q_k) = \sum_{k=1}^K \frac{Q_k}{Q} I(q_k)$$

$$I_G(q_1, q_2) = \underbrace{\left(\frac{3}{6} \times 0\right)}_{I_G(q_1)} + \underbrace{\left(\frac{3}{6} \times \frac{4}{9}\right)}_{I_G(q_2)} = \frac{2}{9} = 0.2222$$



Using Gini Index (cont.)

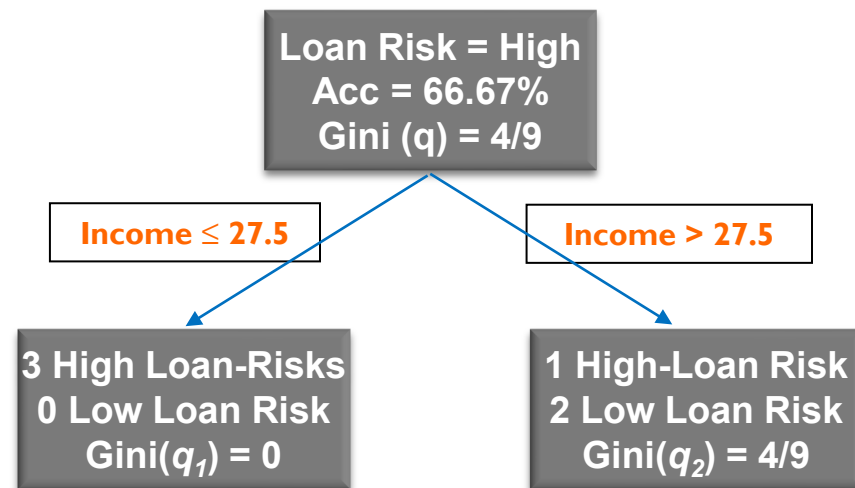
- ▶ Case 2 – Split condition $\text{Income} \leq 32$ versus $\text{Income} > 32$:

$$I_G(q_1, q_2) = \left(\frac{4}{6} \times \frac{3}{8} \right) + \left(\frac{2}{6} \times \frac{1}{2} \right) = \frac{5}{12} = 0.41667$$

- ▶ Case 3 – Split condition $\text{Income} \leq 43$ versus $\text{Income} > 43$:

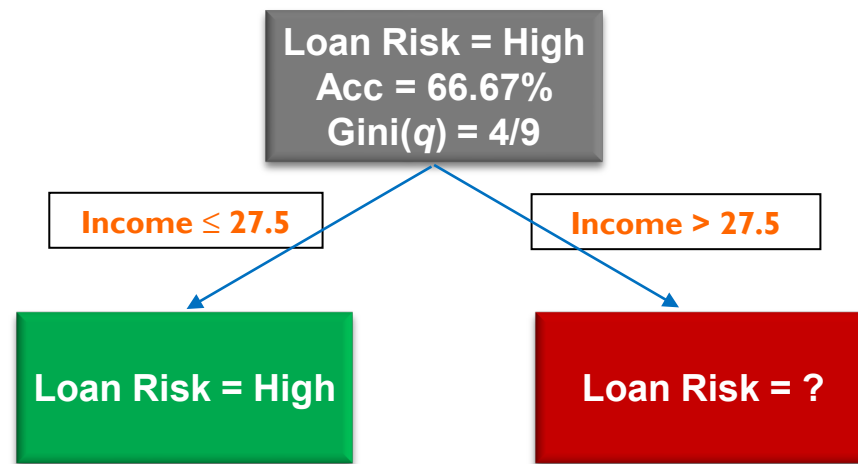
$$I_G(q_1, q_2) = \left(\frac{5}{6} \times \frac{8}{25} \right) + \left(\frac{1}{6} \times 0 \right) = \frac{4}{15} = 0.26667$$

- ▶ Case 1 is the best.
- ▶ Instead of splitting between $\text{Income} \leq 23$ versus $\text{Income} > 23$, the midpoint is selected as actual splitting point: $(23 + 32)/2$.



Using Gini Index (cont.)

- ▶ Apply the tree generating method recursively to nodes that are still not “pure”.



- ▶ Develop a subtree by examining the variable Credit-Rating.
- ▶ Credit-Rating is a discrete variable with ordinal values, i.e., they can be ordered in a meaningful sequence.

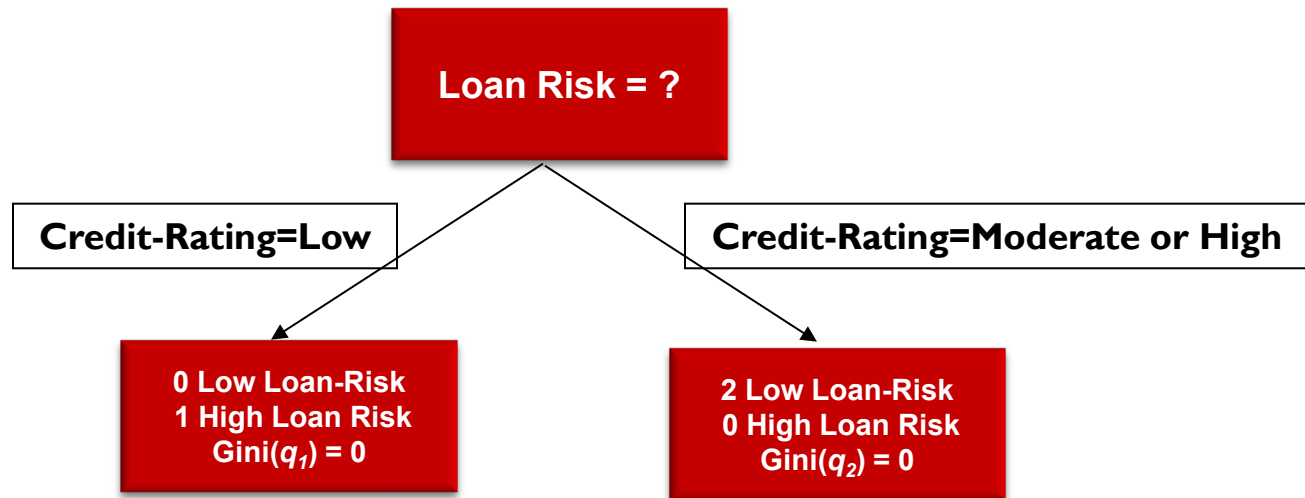
Using Gini Index (cont.)

- ▶ Possible values are {Low, Moderate, High} .
- ▶ Check for best split:
 - ▶ Case 1 – Low versus (Moderate or High)
 - ▶ Case 2 – (Low or Moderate) versus High
- ▶ Compute the Gini index for splitting the node:

Loan Risk = ?

Using Gini Index (cont.)

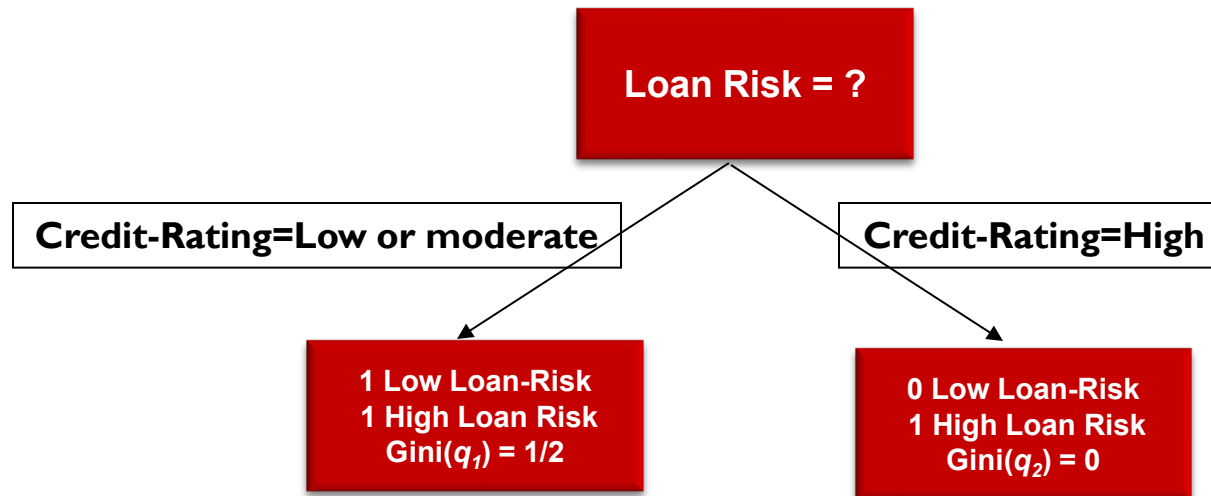
- ▶ Case I – Split Credit-Rating = Low versus Credit-Rating = Moderate or High:



$$I_G(q_1, q_2) = \left(\frac{1}{3} \times 0 \right) + \left(\frac{2}{3} \times 0 \right) = 0$$

Using Gini Index (cont.)

- ▶ Case 2 – Split Credit-Rating = Low or Moderate versus Credit-Rating = High:

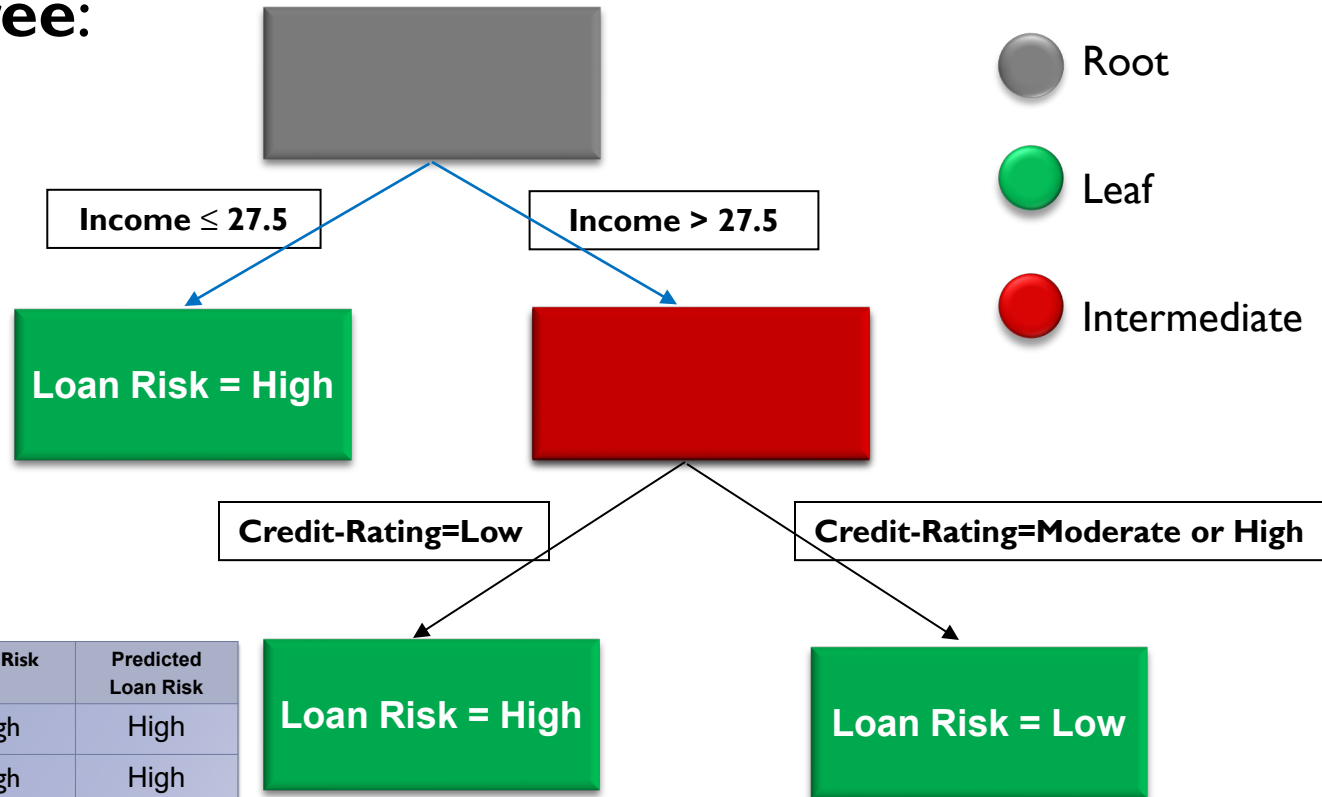


$$I_G(q_1, q_2) = \left(\frac{2}{3} \times \frac{1}{2} \right) + \left(\frac{1}{3} \times 0 \right) = \frac{1}{3}$$

- ▶ Case 2 split is not as good as Case 1 split.

Using Gini Index (cont.)

► Complete tree:



Observation #	Income	Credit Rating	Loan Risk	Predicted Loan Risk
0	23	High	High	High
1	17	Low	High	High
2	43	Low	High	High
3	68	High	Low	Low
4	32	Moderate	Low	Low
5	20	High	High	High

Using Gini Index (cont.)

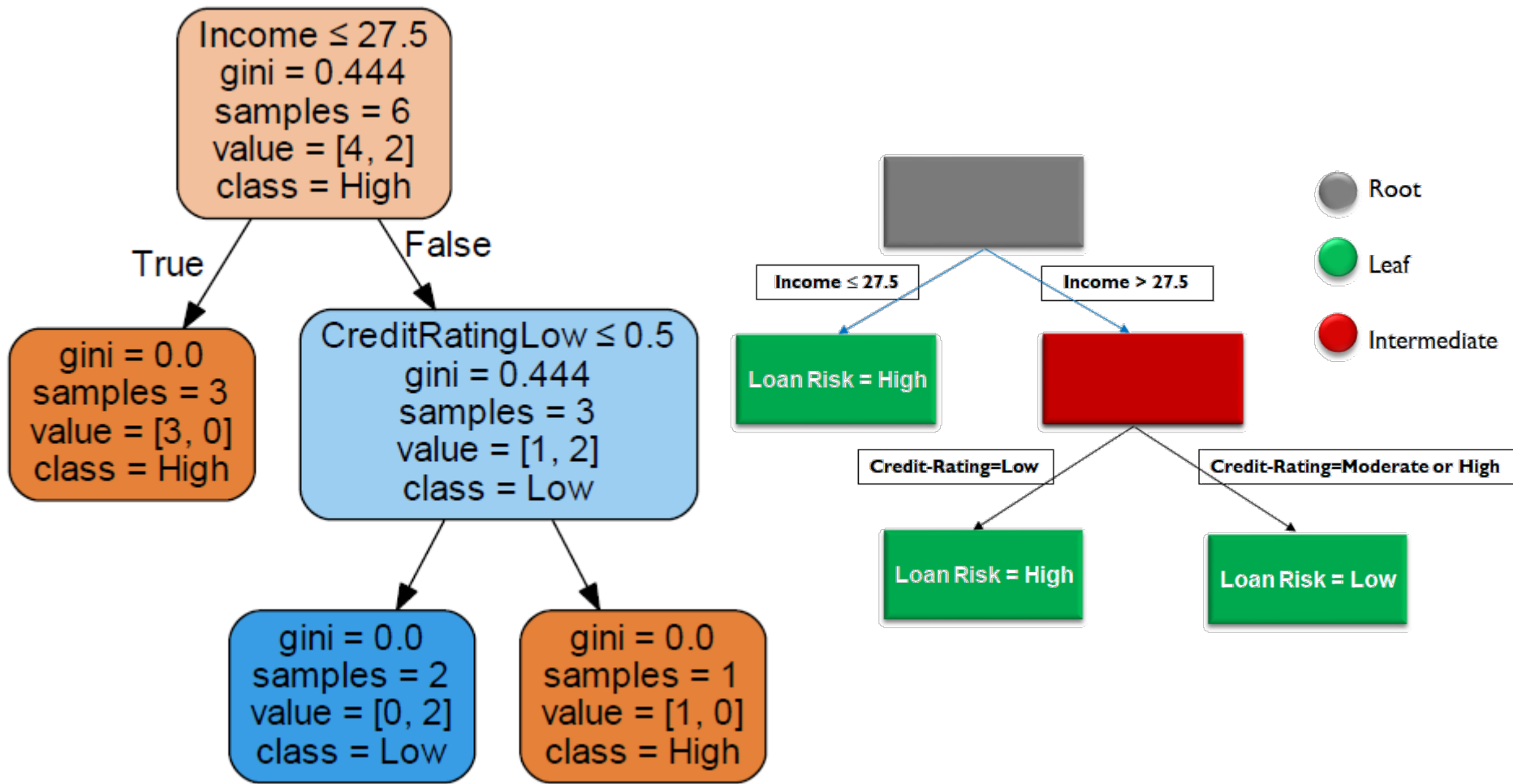
- ▶ The tree achieves 100% accuracy on the training data set.
- ▶ It may over fit the training data instances.
- ▶ Trees may be simplified by **pruning**:
 - ▶ Pre-pruning – Tree growing could be terminated when the number of instances in the node is less than a pre-specified number.
 - ▶ Post-pruning – Removing nodes or branches to improve the accuracy on the test samples.
- ▶ Observe that we have built a binary tree where every non-leaf nodes have 2 branches.

sort of to remove over fitting

Decision Tree in Scikit Learn

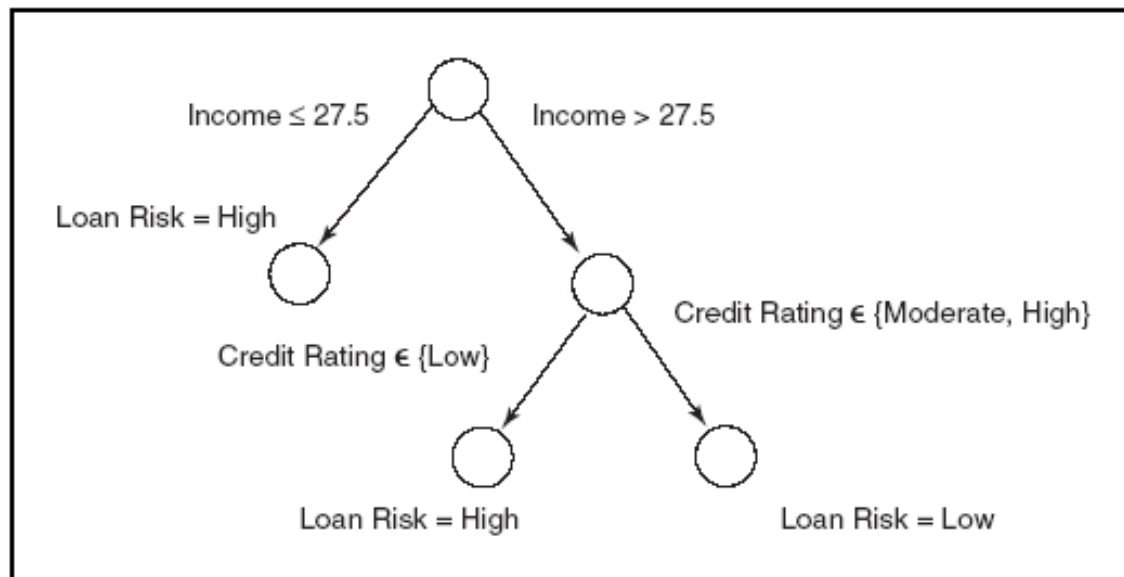
- ▶ We can perform decision tree classification using Scikit Learn's `tree.DecisionTreeClassifier`.
- ▶ However, this class cannot process categorical independent variables and thus we need to recode CreditRating:
 - ▶ Use one hot encoding or one-of-K scheme.
 - ▶ CreditRating has three levels – Low, Moderate and High.
 - ▶ Thus, we will create three binary variables – CreditRatingLow, CreditRatingModerate and CreditRatingHigh.
 - ▶ For each observation, only exactly one of these three variables will be set to 1.
- ▶ Refer to sample source file `src10` for the example.

Decision Tree in Scikit Learn (cont.)



Classification Rule Generation

- ▶ Trace each path from the root node to a leaf node to generate a rule:



If $\text{Income} \leq 27.5$, then $\text{Loan-Risk} = \text{High}$

Else if $\text{Income} > 27.5$ and $\text{Credit-Rating} = \text{Low}$, then $\text{Loan-Risk} = \text{High}$

Else if $\text{Income} > 27.5$ and $\text{Credit-Rating} = \text{Moderate or High}$, then $\text{Loan-Risk} = \text{Low}$



Summary

- ▶ Linear regression can be used to perform regression analysis using one or more independent variables.
- ▶ Categorical independent variables can be used in linear regression through appropriate dummy variable encoding.
- ▶ Decision tree can be used to perform classification analysis.

Q&A





Next Lecture...

- ▶ Learn about:
 - ▶ How to perform prediction with probabilistic classification.
 - ▶ How to perform prediction with advanced classifiers.
 - ▶ How to perform segmentation with clustering.

