National University of Singapore
TCX2002 Introduction to Business Analytics
Tutorial 3

Lesson 5 – Predictive Analytics I – Introduction to Machine Learning and LR

1. Understanding Predictors and Outcome

Scenario: You're predicting monthly sales for a small F&B outlet using a few intuitive predictors.

Setup Data:

```
set.seed(123)

# 36 months: 2022-01 to 2024-12
months <- seq(as.Date("2022-01-01"), as.Date("2024-12-01"), by = "month")
n <- length(months)

# Simple features
Holidays <- ifelse(format(months, "%m") %in% c("01","02","06","11","12"), 1, 0)  #
SG retail-heavy months
Marketing_Spend <- 30000 + 400*(1:n) + 8000*Holidays + rnorm(n, 0, 3000)
Tourist_Arrivals <- 600 + 20*(1:n) + 80*sin(2*pi*(1:n)/12) + rnorm(n, 0, 40)     #
in thousands

# Generate Sales with a simple linear relationship + noise
Sales <- 150000 + 1.8*Marketing_Spend + 220*Tourist_Arrivals + 40000*Holidays +
rnorm(n, 0, 20000)

df <- data.frame(
  Month = months,
  Sales,
  Marketing_Spend,
  Tourist_Arrivals,
  Holidays = factor(Holidays) # as factor for easy interpretation)
```

What are the predictors and outcome (predicted response)?

2. Train-Test Split

Use 2022–2023 as train and 2024 as test.

```
train <- subset(df, Month < as.Date("2024-01-01"))
test  <- subset(df, format(Month, "%Y") == "2024")
nrow(train); nrow(test)
```

National University of Singapore
TCX2002 Introduction to Business Analytics
Tutorial 3

3. Correlation and Multicollinearity Check

Check Correlation among numeric variables:

```
num <- subset(train, select = c(Sales, Marketing_Spend, Tourist_Arrivals))
round(cor(num), 2)
```

VIF to check multicollinearity (needs car package):

```
# install.packages("car") # run once if needed
library(car)
vif(m_mlr)
# Rule of thumb: VIF > 10 may indicate problematic multicollinearity.
```

4. Simple Linear Regression, Multiple Linear Regression & RMSE

```
# Simple LR: Sales ~ Marketing_Spend
m_lr <- lm(Sales ~ Marketing_Spend, data = train)
summary(m_lr)

# Multiple LR: add Tourist_Arrivals and Holidays
m_mlr <- lm(Sales ~ Marketing_Spend + Tourist_Arrivals + Holidays, data = train)
summary(m_mlr)


rmse <- function(a,b) sqrt(mean((a-b)^2))

pred_train_lr  <- predict(m_lr,  train)
pred_train_mlr <- predict(m_mlr, train)
pred_test_lr   <- predict(m_lr,  test)
pred_test_mlr  <- predict(m_mlr, test)

cat("Train RMSE (LR): ", rmse(train$Sales, pred_train_lr),  "\n")
cat("Train RMSE (MLR):", rmse(train$Sales, pred_train_mlr), "\n")
cat("Test RMSE (LR):  ", rmse(test$Sales,  pred_test_lr),   "\n")
cat("Test RMSE (MLR): ", rmse(test$Sales,  pred_test_mlr),  "\n")
```

National University of Singapore
TCX2002 Introduction to Business Analytics
Tutorial 3

Tutorial 3 Learning Outcomes	
1.	**Understand predictors vs outcome and feature representation:** Identify predictors and the outcome. Recognize that categorical predictors create intercept shifts (dummy variables), and see how a simple generative linear process maps to a regression formulation.
2.	**Apply proper model validation with a time-based split and RMSE:** Use a chronological train–test split to avoid leakage in time series–like data. Compute RMSE on both train and test to assess fit and generalization, and interpret gaps between them to detect overfitting or underfitting.
3.	**Diagnose relationships and build/evaluate linear models:** Use correlation matrices to understand linear associations among numeric variables. Check multicollinearity with VIF (flagging concerns when VIF > 10). Fit and interpret Simple versus Multiple LR, interpret coefficients and significance, and compare performance. Expect the MLR to improve test RMSE when additional predictors add true signal, while monitoring multicollinearity and generalization. More in the next lecture.