5주차 과제 예시 답안

```
# Week_5 Quiz

# 문제 바로 아래에 답안을 작성하여, <<r script 형태>>로 저장 후 제출해주시면 됩니다. (.R 파일업로드 / Rproj파일은 업로드하지 않으셔도 됩니다.)
# 답안인 코드만 작성하시면 됩니다. 코드 실행 결과 포함 X
# [R Script Encoding Error] : File > Reopen with Encoding > 'UTF-8' > OK

# 아래 4줄의 코드를 실행 후 문제를 풀어주세요.
# titanic는 수업 > [4주차 과제 외부데이터]에 있습니다.
# 출처 : https://www.kaggle.com/c/titanic/data?select=train.csv
# 데이터 설명 : 타이타닉호의 생존자 분류
library(caret)
library(dplyr)
set.seed(2014
train <- read.csv("경로/titanic_train.csv")  # 경로는 본인의 컴퓨터 환경에 맞게 작성하세요.
ctrl <- trainControl(method = "repeatedcv", number = 5, repeats = 2)



# (5점) 1. train와 데이터의 타겟변수 Survived를 범주형 데이터로 변경해주세요.
# hint. as.factor함수 이용
train$Survived <- as.factor(train$Survived)


# (5점) 2. Pclass는 Ticket 등급을 나타내는 변수입니다. 변수의 고유한 값을 확인할 수 있도록 코드를 작성해주세요.
# hint. unique함수 이용
unique(train$Pclass)


# (5점) 3. Pclass, Age, Fare 변수들의 히스토그램을 한 눈에 볼 수 있도록 시각화 해주세요 (1x3)
# hint. par, hist함수 이용
par(mfrow = c(1,3))
hist(train$Pclass)
hist(train$Age)
hist(train$Fare)


# (5점) 4. train 전체 데이터에 대해 변수산점도를 그려주세요.
# hint. plot함수 이용
plot(train)

# (5점) 5. Age, Fare 변수들을 표준화해주세요.
train$Age <- scale(train$Age)
train$Fare <- scale(train$Fare)

# (5점) 6. train에서 na가 포함된 row는 모두 제거해주세요
# hint. na.omit함수 이용
train_omit <- na.omit(train)


## 아래부터는 6번에서 생성한 데이터를 활용하여 머신러닝 모델을 만들고 해석하는 부분입니다.

모델 실행 결과는 seed에 지정된 값과 실행 횟수에 따라 미세한 차이가 있을 수 있습니다.

# (5점) 7. k를 1~5까지 정의하여 kNN 모델을 생성해주세요. (preProcess 불필요, metric = "Accuracy")
customGrid <- expand.grid(k = 1:5)
knn_fit <- train(Survived ~ .,
                 data = train_omit,
                 method = "knn",
                 trControl = ctrl,
                 tuneGrid = customGrid,
                 metric = "Accuracy")

# (5점) 8. 7번 모델에 대한 실행 결과를 해석해주세요.
# k가 (    )일 때 Accuracy가 (    )%로 가장 높습니다.

k가 1일 때 Accuracy가 56.6%로 가장 높습니다.

knn_fit
---
Resampling: Cross-Validated (5 fold, repeated 2 times)
```

```
Summary of sample sizes: 571, 571, 572, 571, 571, 572, ...
Resampling results across tuning parameters:

  k  Accuracy   Kappa
  1  0.5665321  0.08978845
  2  0.5539249  0.06178985
  3  0.5637496  0.06443303
  4  0.5371073  0.01859827
  5  0.5406087  0.01017729

Accuracy was used to select the optimal model using the largest value.
The final value used for the model was k = 1.
---


# (5점) 9. Boosted Logistic Regression 모델을 생성해주세요. (preProcess 불필요, metric = "Accuracy")

logit_boost_fit <- train(Survived ~ .,
                         data = train_omit,
                         method = "LogitBoost",
                         trControl = ctrl,
                         metric = "Accuracy")


# (5점) 10. 9번 모델에 대한 실행 결과를 해석해주세요.
# nIter가 (    )일 때 Accuracy가 (    )%로 가장 높습니다.

nIter가 31일 때 Accuracy가 78.5%로 가장 높습니다.
logit_boost_fit
---
Resampling: Cross-Validated (5 fold, repeated 2 times)
Summary of sample sizes: 571, 571, 571, 571, 572, 572, ...
Resampling results across tuning parameters:

  nIter  Accuracy   Kappa
  11     0.7814981  0.5436322
  21     0.7737811  0.5261271
  31     0.7857185  0.5510380

Accuracy was used to select the optimal model using the largest value.
The final value used for the model was nIter = 31.
---

# (5점) 11. Naive Bayes 모델을 생성해주세요. (preProcess 불필요, metric = "Accuracy")

nb_fit <- train(Survived ~ .,
                data = train_omit,
                method = "naive_bayes",
                trControl = ctrl,
                metric = "Accuracy")


# (5점) 12. 11번 모델에 대한 실행 결과를 해석해주세요.
# useKernel이 (    )일 때 Accuracy가 (    )%로 가장 높습니다.

useKernel이 TRUE일 때 Accuracy가 59.3%로 가장 높습니다.

nb_fit
---
No pre-processing
Resampling: Cross-Validated (5 fold, repeated 2 times)
Summary of sample sizes: 572, 571, 571, 571, 571, 571, ...
Resampling results across tuning parameters:

  usekernel  Accuracy   Kappa
  FALSE      0.4061657  0
   TRUE      0.5938343  0
---


# (5점) 13. 연속형 숫자 피쳐만 선택하여 주성분 분석을 진행해주세요
# hint. 연속형 숫자 피쳐 = 데이터 타입이 int와 num인 변수
# hint. prcomp 함수 활용

num_feature <- c("PassengerId","Pclass","Age","SibSp","Parch","Fare")
num_data <- train_omit[, num_feature]
pca_num <- prcomp(num_data)
```

```
PassengerId, Pclass : 2개 변수는 수치형 보다는 범주형으로 해석하는게 문제에 적합할 수 있습니다. (PassengerId,Pclass 포함/제외 모두 정답 처리)
변수를 어떤 데이터 타입으로 정의하여 모델링을 하는지에 따라 모델의 성능이 달라질 수 있는 점 참고 부탁드립니다.


# (5점) 14. 13번 모델의 summary 결과를 해석해주세요.
# 축이 4개 일 때 전체 변동성의 (    )를 설명합니다.

summary(pca_num)
축이 4개 일 때 전체 변동성의 99%를 설명합니다.

Importance of components:
                            PC1     PC2     PC3     PC4     PC5   PC6
Standard deviation     259.1195 1.25750 1.19431 0.79248 0.69325 0.552
Proportion of Variance   0.9999 0.00002 0.00002 0.00001 0.00001 0.000
Cumulative Proportion    0.9999 0.99996 0.99998 0.99999 1.00000 1.000

전체 변동성은 "Cumulative Proportion" 를 의미합니다.
```