#

# multiple linear regression

#

# numeric predictors

fit <- lm(Fertility ~ Examination + Catholic, data = swiss)

summary(fit)

fit2 <- lm(Fertility ~ Examination\*Catholic, data = swiss)

summary(fit2)

confint(fit2)

# categorical predictors

hist(swiss$Catholic, col = 'red')

swiss$religious <- ifelse(swiss$Catholic > 60, 'Lots', 'Few')

swiss$religious <- as.factor(swiss$religious)

fit3 <- lm(Fertility ~ Examination + religious, data = swiss)

summary(fit3)

fit4 <- lm(Fertility ~ religious\*Examination, data = swiss)

summary(fit4)

# plots

ggplot(swiss, aes(x = Examination, y = Fertility)) +

geom\_point()

ggplot(swiss, aes(x = Examination, y = Fertility)) +

geom\_point() +

geom\_smooth()

ggplot(swiss, aes(x = Examination, y = Fertility)) +

geom\_point() +

geom\_smooth(method = 'lm')

ggplot(swiss, aes(x = Examination, y = Fertility, col = religious)) +

geom\_point()

ggplot(swiss, aes(x = Examination, y = Fertility, col = religious)) +

geom\_point() +

geom\_smooth()

ggplot(swiss, aes(x = Examination, y = Fertility, col = religious)) +

geom\_point() +

geom\_smooth(method = 'lm')

#

fit5 <- lm(Fertility ~ religious\*Infant.Mortality\*Examination, data = swiss)

summary(fit5)

# model comparison

rm(swiss)

swiss <- data.frame(swiss)

fit\_full <- lm(Fertility ~ ., data = swiss)

summary(fit\_full)

fit\_reduced1 <- lm(Fertility ~ Infant.Mortality + Examination + Catholic + Education, data = swiss)

summary(fit\_reduced1)

anova(fit\_full, fit\_reduced1)

fit\_reduced2 <- lm(Fertility ~ Infant.Mortality + Education + Catholic + Agriculture, data = swiss)

summary(fit\_reduced2)

anova(fit\_full, fit\_reduced2)

# model selection

optimal\_fit <- step(fit\_full, direction = 'backward')

summary(optimal\_fit)

#

# regression diagnostics

#

library(ggplot2)

data(swiss)

str(swiss)

# relationships between all variables

pairs(swiss)

ggplot(swiss, aes(x = Examination, y = Education)) +

geom\_point()

# Outliers

ggplot(swiss, aes(x = Examination, y = Education)) +

geom\_point() +

geom\_smooth(method = 'lm')

# Normality of variables distributions

ggplot(swiss, aes(x = Examination)) +

geom\_histogram()

ggplot(swiss, aes(x = Education)) +

geom\_histogram()

# linearity

ggplot(swiss, aes(x = Examination, y = Education)) +

geom\_point() +

geom\_smooth()

lm1 <- lm(Education ~ Examination, swiss)

summary(lm1)

swiss$Examination\_squared <- (swiss$Examination)^2

lm2 <- lm(Education ~ Examination + Examination\_squared, swiss)

summary(lm2)

anova(lm2, lm1)

swiss$lm1\_fitted <- lm1$fitted

swiss$lm2\_fitted <- lm2$fitted

swiss$lm1\_resid <- lm1$resid

swiss$lm2\_resid <- lm2$resid

swiss$obs\_number <- 1:nrow(swiss)

ggplot(swiss, aes(x = Examination, y = Education)) +

geom\_point(size = 3) +

geom\_line(aes(x = Examination, y = lm1\_fitted), col = 'red', lwd=1) +

geom\_line(aes(x = Examination, y = lm2\_fitted), col = 'blue', lwd=1)

ggplot(swiss, aes(x = lm1\_fitted, y = lm1\_resid)) +

geom\_point(size = 3) + geom\_hline(y=0, col = 'red', lwd = 1)

ggplot(swiss, aes(x = lm2\_fitted, y = lm2\_resid)) +

geom\_point(size = 3) + geom\_hline(y=0, col = 'red', lwd = 1)

# independence of errors

ggplot(swiss, aes(x = obs\_number, y = lm1\_resid)) +

geom\_point(size = 3) + geom\_smooth()

ggplot(swiss, aes(x = obs\_number, y = lm2\_resid)) +

geom\_point(size = 3) + geom\_smooth()

# Homoscedasticity

ggplot(swiss, aes(x = lm1\_fitted, y = lm1\_resid)) +

geom\_point(size = 3)

ggplot(swiss, aes(x = lm2\_fitted, y = lm2\_resid)) +

geom\_point(size = 3)

# Errors Normally distributed

ggplot(swiss, aes(x = lm1\_resid)) +

geom\_histogram(binwidth = 4, fill = 'white', col = 'black')

qqnorm(lm1$residuals)

qqline(lm1$residuals)

shapiro.test(lm1$residuals)

ggplot(swiss, aes(x = lm2\_resid)) +

geom\_histogram(binwidth = 4, fill = 'white', col = 'black')

qqnorm(lm2$residuals)

qqline(lm2$residuals)

shapiro.test(lm2$residuals)

library(ggplot2)

my\_df <- read.csv("train.csv", sep=";")

str(my\_df)

ggplot(my\_df, aes(read, math, col = gender))+

geom\_point(size = 5)+

facet\_grid(.~hon)+

theme(axis.text=element\_text(size=25),

axis.title=element\_text(size=25,face="bold"))

fit <- glm(hon ~ read + math + gender, my\_df, family = "binomial")

summary(fit)

exp(fit$coefficients)

head(predict(object = fit))

head(predict(object = fit, type = "response"))

my\_df$prob <- predict(object = fit, type = "response")

library(ROCR)

pred\_fit <- prediction(my\_df$prob, my\_df$hon)

perf\_fit <- performance(pred\_fit,"tpr","fpr")

plot(perf\_fit, colorize=T , print.cutoffs.at = seq(0,1,by=0.1))

auc <- performance(pred\_fit, measure = "auc")

str(auc)

perf3 <- performance(pred\_fit, x.measure = "cutoff", measure = "spec")

perf4 <- performance(pred\_fit, x.measure = "cutoff", measure = "sens")

perf5 <- performance(pred\_fit, x.measure = "cutoff", measure = "acc")

plot(perf3, col = "red", lwd =2)

plot(add=T, perf4 , col = "green", lwd =2)

plot(add=T, perf5, lwd =2)

legend(x = 0.6,y = 0.3, c("spec", "sens", "accur"),

lty = 1, col =c('red', 'green', 'black'), bty = 'n', cex = 1, lwd = 2)

abline(v= 0.225, lwd = 2)

my\_df$pred\_resp <- factor(ifelse(my\_df$prob > 0.225, 1, 0), labels = c("N", "Y"))

my\_df$correct <- ifelse(my\_df$pred\_resp == my\_df$hon, 1, 0)

ggplot(my\_df, aes(prob, fill = factor(correct)))+

geom\_dotplot()+

theme(axis.text=element\_text(size=25),

axis.title=element\_text(size=25,face="bold"))

mean(my\_df$correct)

test\_df <- read.csv("test.csv", sep = ";")

test\_df$hon <- NA

test\_df$hon <- predict(fit, newdata = test\_df, type = "response")

View(test\_df)