#Linear regression with simple data

#Linear regression with Wine Data

#Linear regression with Money baseball

#Linear regression with NBA

**#The CPI Data**

year <- rep(2008:2010, each = 4)

quarter <- rep(1:4, 3)

cpi <- c(162.2, 164.6, 166.5, 166, 166.2, 167, 168.6, 169.5, 171,172.1, 173.3, 174)

plot(cpi, xaxt = "n", ylab = "CPI", xlab = "")

# draw x-axis, where 'las=3' makes text vertical

axis(1, labels = paste(year, quarter, sep = "Q"), at = 1:12, las = 3)

**#Linear Regression**

# correlation between CPI and year / quarter

cor(year, cpi)

## [1] 0.9096316

cor(quarter, cpi)

## [1] 0.3738028

## build a linear regression model with function lm()

fit <- lm(cpi ~ year + quarter)

fit

##

## Call:

## lm(formula = cpi ~ year + quarter)

##

## Coefficients:

## (Intercept) year quarter

## -7644.488 3.888 1.167

**## Predection**

##What will the CPI be in 2011?

cpi2011 <- fit$coefficients[[1]] + fit$coefficients[[2]] \* 2011 + fit$coefficients[[3]] \* (1:4)

cpi2011

## [1] 174.4417 175.6083 176.7750 177.9417

#More details of the model can be obtained with the code below.

**attributes(fit)**

## $names

## [1] "coefficients" "residuals" "effects"

## [4] "rank" "fitted.values" "assign"

## [7] "qr" "df.residual" "xlevels"

## [10] "call" "terms" "model"

##

## $class

## [1] "lm"

fit$coefficients

## (Intercept) year quarter

## -7644.487500 3.887500 1.166667

**#3D Plot of the Fitted Model**

install.packages("scatterplot3d") # if you have already installed just load it

library(scatterplot3d)

s3d <- scatterplot3d(year, quarter, cpi, highlight.3d = T, type = "h",lab = c(2, 3)) # lab: number of tickmarks on x-/y-axes

s3d$plane3d(fit) # draws the fitted plane

**#Prediction of CPIs in 2011**

data2011 <- data.frame(year = 2011, quarter = 1:4)

cpi2011 <- predict(fit, newdata = data2011)

style <- c(rep(1, 12), rep(2, 4))

plot(c(cpi, cpi2011), xaxt = "n", ylab = "CPI", xlab = "", pch = style,col = style)

axis(1, at = 1:16, las = 3, labels = c(paste(year, quarter, sep = "Q"),"2011Q1", "2011Q2", "2011Q3", "2011Q4"))

**#Generalized Linear Model (GLM)**

#Unifies various other statistical models, including linearregression, logistic regression and Poisson regression

data("bodyfat", package="TH.data")

myFormula <- DEXfat ~ age + waistcirc + hipcirc + elbowbreadth +kneebreadth

bodyfat.glm <- glm(myFormula, family = gaussian("log"), data = bodyfat)

summary(bodyfat.glm)

**#Prediction with Generalized Linear Regression Model**

pred <- predict(bodyfat.glm, type = "response")

plot(bodyfat$DEXfat, pred, xlab = "Observed", ylab = "Prediction")

abline(a = 0, b = 1)

#Linear regression with Wine Data

# VIDEO 4

# Read in data

wine = read.csv("wine.csv")

str(wine)

summary(wine)

# Linear Regression (one variable)

model1 = lm(Price ~ AGST, data=wine)

summary(model1)

# Sum of Squared Errors

model1$residuals

SSE = sum(model1$residuals^2)

SSE

# Linear Regression (two variables)

model2 = lm(Price ~ AGST + HarvestRain, data=wine)

summary(model2)

# Sum of Squared Errors

SSE = sum(model2$residuals^2)

SSE

# Linear Regression (all variables)

model3 = lm(Price ~ AGST + HarvestRain + WinterRain + Age + FrancePop, data=wine)

summary(model3)

# Sum of Squared Errors

SSE = sum(model3$residuals^2)

SSE

# VIDEO 5

# Remove FrancePop

model4 = lm(Price ~ AGST + HarvestRain + WinterRain + Age, data=wine)

summary(model4)

# VIDEO 6

# Correlations

cor(wine$WinterRain, wine$Price)

cor(wine$Age, wine$FrancePop)

cor(wine)

# Remove Age and FrancePop

model5 = lm(Price ~ AGST + HarvestRain + WinterRain, data=wine)

summary(model5)

# VIDEO 7

# Model Validation with the test set.

# Read in test set

wineTest = read.csv("wine\_test.csv")

str(wineTest)

# Make test set predictions

predictTest = predict(model4, newdata=wineTest)

predictTest

# Compute R-squared

SSE = sum((wineTest$Price - predictTest)^2)

SST = sum((wineTest$Price - mean(wine$Price))^2)

1 - SSE/SST

#Linear regression with Money baseball

# VIDEO 2

# Read in data

baseball = read.csv("baseball.csv")

str(baseball)

# Subset to only include moneyball years

moneyball = subset(baseball, Year < 2002)

str(moneyball)

# Compute Run Difference

moneyball$RD = moneyball$RS - moneyball$RA

str(moneyball)

# Scatterplot to check for linear relationship

plot(moneyball$RD, moneyball$W)

# Regression model to predict wins

WinsReg = lm(W ~ RD, data=moneyball)

summary(WinsReg)

# VIDEO 3

str(moneyball)

# Regression model to predict runs scored

RunsReg = lm(RS ~ OBP + SLG + BA, data=moneyball)

summary(RunsReg)

RunsReg = lm(RS ~ OBP + SLG, data=moneyball)

summary(RunsReg)

#Linear regression with NBA

# VIDEO 1

# Read in the data

NBA = read.csv("NBA\_train.csv")

str(NBA)

# VIDEO 2

# How many wins to make the playoffs?

table(NBA$W, NBA$Playoffs)

# Compute Points Difference

NBA$PTSdiff = NBA$PTS - NBA$oppPTS

# Check for linear relationship

plot(NBA$PTSdiff, NBA$W)

# Linear regression model for wins

WinsReg = lm(W ~ PTSdiff, data=NBA)

summary(WinsReg)

# VIDEO 3

# Linear regression model for points scored

PointsReg = lm(PTS ~ X2PA + X3PA + FTA + AST + ORB + DRB + TOV + STL + BLK, data=NBA)

summary(PointsReg)

# Sum of Squared Errors

PointsReg$residuals

SSE = sum(PointsReg$residuals^2)

SSE

# Root mean squared error

RMSE = sqrt(SSE/nrow(NBA))

RMSE

# Average number of points in a season

mean(NBA$PTS)

# Remove insignifcant variables

summary(PointsReg)

PointsReg2 = lm(PTS ~ X2PA + X3PA + FTA + AST + ORB + DRB + STL + BLK, data=NBA)

summary(PointsReg2)

PointsReg3 = lm(PTS ~ X2PA + X3PA + FTA + AST + ORB + STL + BLK, data=NBA)

summary(PointsReg3)

PointsReg4 = lm(PTS ~ X2PA + X3PA + FTA + AST + ORB + STL, data=NBA)

summary(PointsReg4)

# Compute SSE and RMSE for new model

SSE\_4 = sum(PointsReg4$residuals^2)

RMSE\_4 = sqrt(SSE\_4/nrow(NBA))

SSE\_4

RMSE\_4

# Model Validation with the test set.

# VIDEO 4

# Read in test set

NBA\_test = read.csv("NBA\_test.csv")

# Make predictions on test set

PointsPredictions = predict(PointsReg4, newdata=NBA\_test)

# Compute out-of-sample R^2

SSE = sum((PointsPredictions - NBA\_test$PTS)^2)

SST = sum((mean(NBA$PTS) - NBA\_test$PTS)^2)

R2 = 1 - SSE/SST

R2

# Compute the RMSE

RMSE = sqrt(SSE/nrow(NBA\_test))

RMSE