Alex Hyman-FIN654-PS01

Alex Hyman

7/4/2018

Practice set 1: present value

```
#Defining all of our given variables
WACC <- 0.18
sales.rate <- c(0.5, 0.5, 0.5, 0.15)
cost.rate <- 0.75
depreciation.rate <- 0.05
tax.rate <- 0.35
sales.current <- 80
years <- 1:4

# Working capital needs to increase 15% each year
# Need an increase of fixed assets at the rate of 10% of sales
# Working capital = $10, NFA = $90, accumulated depreciation = $15
WC <- 10
NFA <- 90
depreciation.accumulated <- 15

# Projecting sales, cost, increments to NFA, Increments to WC, depreciation, taxes, and FCF for next 4
# Modeling Sales
sales <- rep(0, 4)
i <- 1
for (rate in sales.rate){
    if (i == 1){
        sales[i] <- sales.current * (1 + sales.rate[i])
        i = i + 1
    } else{
        sales[i] <- sales[i-1] * (1 + sales.rate[i])
        i = i + 1
    }
}

# Modeling Cost
# Cost is 75% of sales
cost <- sales * cost.rate

# Increments to NFA
# NFA is increased at 10% of sales
NFA.incr <- sales * 0.1

# Increments to working capital
# 15% increase each year
WC.incr <- WC * (1 + 0.15)^years
WC.incr <- c(WC, WC.incr)
WC.incr <- WC.incr[2:5] - WC.incr[1:4]
```

```r
# Modeling depreciation
# depreciation is always 5% of NFA
NFA <- NFA.incr + NFA
depreciation <- NFA * depreciation.rate

# Modeling taxes
# Taxes are based on sales - cost - depreciation
income.taxable <- sales - cost - depreciation
tax <- income.taxable * tax.rate

# Free Cash Flow
CF.operational <- income.taxable + depreciation - tax
CF.capital <- NFA.incr + depreciation
CF.wc <- WC.incr
FCF <- CF.operational - CF.capital - CF.wc

# Forming tables of results
tables.names <- c("Sales", "Cost", "Working Capital (incr.)", "Net Fixed Assets (incr.)", "Free Cash Fl
# Assign Projection Labels
table.year <- years
table.data <- rbind(sales, cost, WC.incr, NFA.incr, FCF)
rownames(table.data) <- tables.names
colnames(table.data) <- table.year
knitr::kable(table.data)
```

	1	2	3	4
Sales	120.000	180.000	270.00000	310.500000
Cost	90.000	135.000	202.50000	232.875000
Working Capital (incr.)	1.500	1.725	1.98375	2.281312
Net Fixed Assets (incr.)	12.000	18.000	27.00000	31.050000
Free Cash Flow	2.685	6.015	11.08875	13.190812

Practice set 2: regression

1. Build a table that explores this data set variable by variable and relationships among variables.

```r
# Creating labels for the tables
table.outputs <- c("Expenses", "Payroll", "Full-Time Employees")
table.inputs <- c("Admissions", "Outpatients")

# Reading the data
x.data <- read.csv("data/hospitals.csv")

# Creating an X matrix of the inputs
X <- cbind(x.data[["admissions"]], x.data[["outpatients"]])

# Separating the various output columns into different matrixes
expense <-matrix(x.data[["expense"]], ncol = 1)
payroll <- matrix(x.data[["payroll"]], ncol = 1)
fte <- matrix(x.data[["fte"]], ncol = 1)
```

2. Investigate the influence of admission and outpatient rates on expenses and payroll. First, form these

arrays.

```r
# (XTT)-1XTt <- beta.hat
XTX.Inverse <- solve(t(X) %*% X)
#regression of expense
beta.hat.expense <- XTX.Inverse %*% t(X) %*% expense
#regression of payroll
beta.hat.payroll <- XTX.Inverse %*% t(X) %*% payroll
#regression of fte
beta.hat.fte <- XTX.Inverse %*% t(X) %*% fte

#Creating a table of the various regression coefficients
table.data <- rbind(t(beta.hat.expense), t(beta.hat.payroll), t(beta.hat.fte))
rownames(table.data) <- table.outputs
colnames(table.data) <- table.inputs
beta.hat <- beta.hat.expense
y <- expense
knitr::kable(table.data)
```

	Admissions	Outpatients
Expenses	8.6924580	0.0795136
Payroll	3.6837693	0.0507119
Full-Time Employees	0.1005839	0.0013529

3. Use this code to investigate further the relationship among predicted expenses and the drivers, admissions and outpatients.

```r
# Loading the packages
require(reshape2)
```

```
## Loading required package: reshape2
```

```r
require(ggplot2)
```

```
## Loading required package: ggplot2
```

```r
# Setting up the residual plot
actual <- y
predicted <- X %*% beta.hat

# Residuals = Actual - predicted
residual <- actual - predicted

# Creating a data frame for the actual, predicted, and residual
results <- data.frame(actual = actual, predicted = predicted, residual = residual)

# Selecting the minimum and maximum for the plot limits
min_xy <- min(min(results$actual), min(results$predicted))
max_xy <- max(max(results$actual), max(results$predicted))

# Creating a data frame with the columns predicted, variable (factor: actual, residual), and the
# value of either the residual or the actual
plot.melt <- melt(results, id.vars = "predicted")

# Adding the min and max data points to the data frame
plot.data <- rbind(plot.melt, data.frame(predicted = c(min_xy, max_xy), variable = c("actual",
```

```
    "actual"), value = c(max_xy, min_xy)))

# Plotting the plot
p <- ggplot(plot.data, aes(x = predicted, y = value)) + geom_point(size = 2.5) + theme_bw()
p <- p + facet_wrap(~variable, scales = "free")
p
```