# Compute the average authorized ROE using the rp method.

# Regression of ERP on interest rates.

# Regression of unlevered ROE on interest rates.

# Write the .csv file for Excel use

# CREATED BY: F PAMPUSH

# CREATED ON: 4/29/2024

# MODIFIED ON: 07/12/2024

library(tidyverse)

library(readxl)

library(ggpmisc)

library(ggrepel)

# Color palette for plots.

colPalette <- c("#156082", "#E97132","#196B24", "#0F9ED5", "#A02B93",

"#4EA72E", "#E69F00", "#CC79A7")

# Load US and Canadian authorized return data downloaded from SNL

rDir<-"C:/Users/frank/OneDrive/Ontario Energy Board/R\_analysis/Rdata"

rawDir<-"C:/Users/frank/OneDrive/Ontario Energy Board/R\_analysis/Raw"

# Load the SNL utility authorized return data.

load(paste(rDir,"usSNL.Rdata",sep="/"))

load(paste(rDir,"caSNL.Rdata",sep="/"))

# Load the interest rate data.

load(paste(rDir,"CA10.Rdata",sep="/"))

load(paste(rDir,"DGS.Rdata",sep="/"))

load(paste(rDir,"DBAA.Rdata",sep="/"))

# Load the corporate tax rate data

cit<-read\_xlsx(path=paste(rawDir,"CorporateTaxRates.xlsx",sep="/"),

sheet="Corporate\_Tax\_Rates",skip=3)

names(cit)<-c("yrAuth","cait","usit")

# Prepare the US data SNL.

# Eliminate Limited Issue Riders

# Eliminate Illinois

# Keep only Authorized ROEs>0.

DERatioDeemed<-as.data.frame((c(1,1.5,1.857))) # Canada debt:equity ratio

names(DERatioDeemed)<-"DERatio"

usfile2<-usfile%>%

mutate(ym1=as.POSIXct(ymd(paste(year(AuthDate),month(AuthDate),"01",sep="-"))))%>%

left\_join(cit,by="yrAuth")%>%

left\_join(DGS,by=c("AuthDate"="observation\_date"))%>%

left\_join(CA10,by=c("ym1"="observation\_date"))%>%

filter(!(state=="IL" & service=="Electric"),

service=="Electric",

reqROE>0,

AuthROE>0,

ReqEqRatio>0,

AuthERatio>0,

CaseType!="Limited-Issue Rider")%>%

# Create unlevered & relevered ROEs

mutate(fileDate=ymd(fileDate),

AuthDate=ymd(AuthDate),

authROE\_1=AuthROE/100,

authE2cap\_1=AuthERatio/100,

dgs30\_1=DGS30/100,

#ca10\_1=CA10/100,

DERatio=(1/(authE2cap\_1))-1,

ROEU\_1=((authROE\_1+DERatio\*(1-uscitr)\*(dgs30\_1))/

(1+DERatio\*(1-uscitr))),

ROEU=ROEU\_1\*100,

ROEL100=100\*(ROEU\_1+DERatioDeemed[1,1]\*(1-cait)\*(ROEU\_1-dgs30\_1)),

ROEL150=150\*(ROEU\_1+DERatioDeemed[2,1]\*(1-cait)\*(ROEU\_1-dgs30\_1)),

ROEL185=100\*(ROEU\_1+DERatioDeemed[3,1]\*(1-cait)\*(ROEU\_1-dgs30\_1)))%>%

# Simplifiy the file by removing unused columns

select(-caseid,-rateChange,-reqRB,

-RateChange,-RBValMethod,-Lag,-Interim,

-Phase,-DecisionType,-RB2,-TestYrEnd,-uscitr)%>%

left\_join(DBAA,by=c("AuthDate"="observation\_date"))%>%

mutate(DBAAdiff=DBAA-DGS30)

#ytm<-data.frame(t(c(4.535,3.351,5.79))) # 30 year bond 6/24/2024

ytm<-data.frame(t(c(4.06,3.351,5.79))) # 30 year bond 6/24/2024

names(ytm)<-c("DGS30","CA10","DBAA")

ytm<-ytm%>%

mutate(DBAAdiff=DBAA-DGS30)

cit24<-cit%>%filter(yrAuth==2024)

# Compute the regression line ERP~a + b rf and unleveredROE~a+b rf \_ DBaadiff.

lm\_data<-usfile2%>%

filter(ROEU>0,

ROEL150>0,

DGS30>0)%>%

mutate(ERPu=ROEU-DGS30)#,

# fileDate=as.Date(fileDate),

# AuthDate=as.Date(AuthDate))

#lmERPu<-lm(ERPu~DGS30+DBAAdiff,data=lm\_data)

lmERPu<-lm(ERPu~DGS30+DBAA,data=lm\_data)

summary(lmERPu)

lmROEu<-lm(ROEU~DGS30+DBAA,data=lm\_data)

summary(lmROEu)

# Create a prediction based on LEI specification model.

newdata1<- data.frame(tax= rep(0.265,6),

DGS30=c(rep(ytm[1,1]/100,4),0.034,ytm[1,1]/100),

DBAA=rep(ytm[1,3]/100,6))

newdata1<-newdata1%>%

mutate(DBAAdiff=DBAA-DGS30)

ROR\_hat<-data.frame(ERPu=predict(lmERPu,newdata=newdata1\*100)/100)

ROR\_hat<-ROR\_hat%>%

mutate(ROEu=as.numeric(predict(lmROEu,newdata=newdata1\*100)/100),

ROEL100=(ROEu+1\*(1-newdata1$tax)\*(ROEu-newdata1$DGS30)),

ROEL150=(ROEu+1.5\*(1-newdata1$tax)\*(ROEu-newdata1$DGS30)),

ROEL185=(ROEu+1.857\*(1-newdata1$tax)\*(ROEu-newdata1$DGS30)),

scenario=paste("roe",round(newdata1$DGS30\*100,2),sep="\_"),

tax=newdata1$tax,

DGS30=newdata1$DGS30)%>%

cbind(newdata1[,3:4])%>%

mutate(rmseE=rep(sqrt(mean(lmERPu$residuals^2)),nrow(newdata1)),

rmseR=rep(sqrt(mean(lmROEu$residuals^2)),nrow(newdata1)),

rmseE=rmseE/100,

rmseR=rmseR/100)