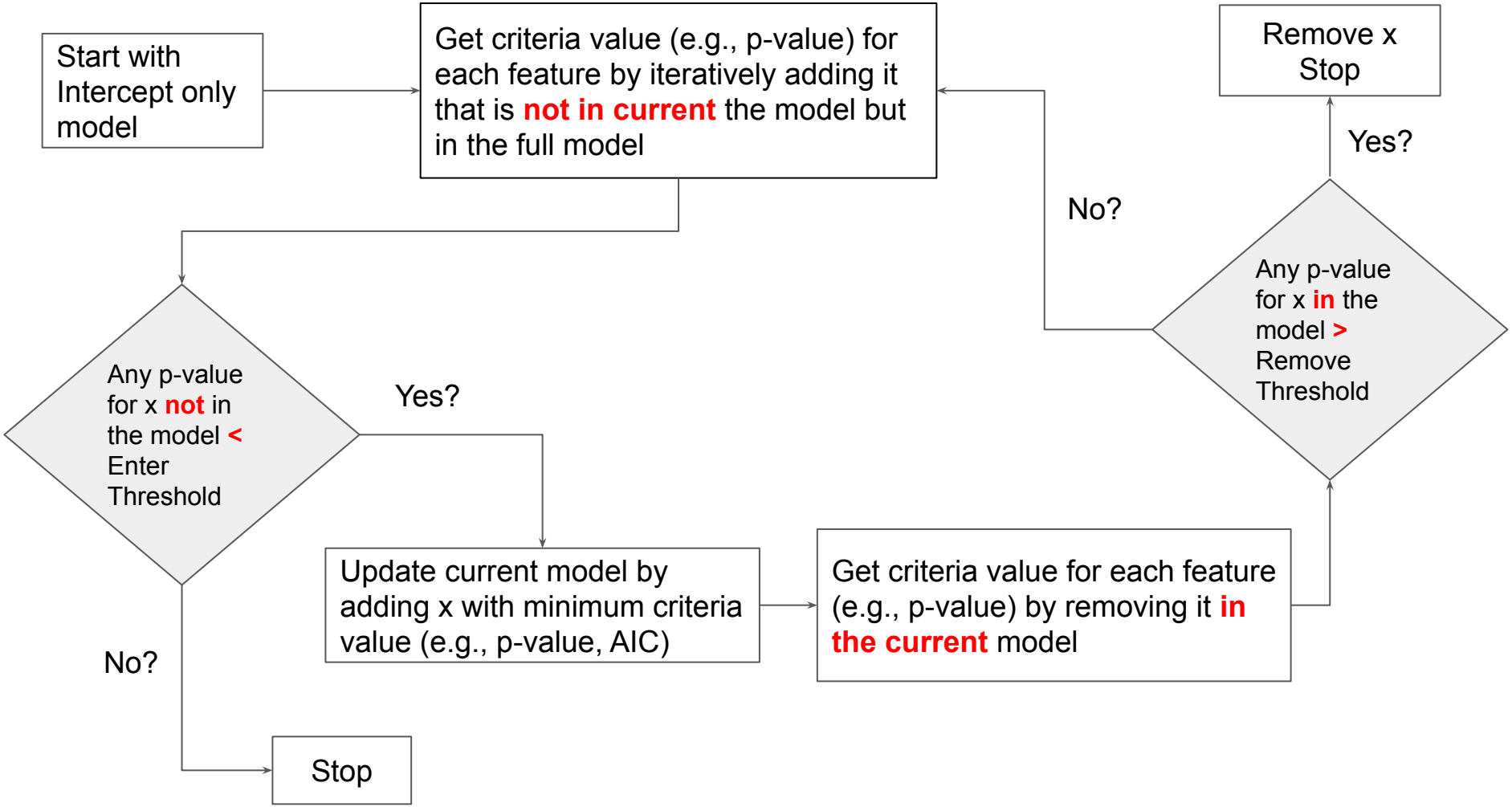
Stepwise 구현

Roh


Start with Intercept only model
Get criteria value (e.g., p-value) for each feature by iteratively adding it that is not in current the model but in the full model
Any p-value for x not in the model < Enter Threshold
Yes?
No?
Update current model by adding x with minimum criteria value (e.g., p-value, AIC)
Get criteria value for each feature (e.g., p-value) by removing it in the current model
Any p-value for x in the model > Remove Threshold
Yes?
No?
Remove x
Stop
Stop

주의할 점 (1)

기준을 무엇으로 할것인가? 정해진 답이 없음!

- Test Statistics and its P-value
  - T test (x)
  - LR test
  - Rao Score test
  - Wald test
- R2
- AIC / BIC

SAS

추가 Criteria: Rao Score

제거 Criteria: Wald Score

Note: Wald >= LR >= Rao

주의할 점 (2)

Regression 결과에 나오는 변수별 p-value를 쓰는게 아니다!

t-test statistic 과 해당하는 p-value는 beta = 0 을 테스트 하는 것

Stepwise selection에서 하고 싶은것은 모델간의 비교이다

Intercept only : $Y = b0$

Step 1 Model : $Y = b0 + b1\ X1$

...

Full model : $Y = b0 + b1\ X1 + b2\ X2 + \ldots + bk\ Xk$

다시말해서,

새로운 변수를 추가 및 삭제 했을때

- Residual Sum of Square (OLS)
- Likelihood (LOGIT)

이 현재모델과 추가삭제한 모델이 000을 기준으로

Significant 하게 달라졌는지 체크!

AIC/BIC 그리고 R2는 비교

나머지 LR/Rao/Wald 는 테스트

주의할 점 (4)

- Categorical Variable 인경우
  - 변수를 추가하거나 뺄때 더미코딩 되어있는 각각의 변수가 아니라 전체를 한번에 추가하거나 빼야함
  - 과도한 Linear Dependency (Multicollinearity) 가 있을 수 있음
    - 미리 제거 필요!

- 자동으로 선택된 변수 set을 과도하게 믿게됨
  - Knowledge of the subject expert 가 필요
    - Forced In (사업적으로 반드시 필요한 경우)
    - Forced Out (Nice에서 더이상 제공을 안해주는 경우)
  - 에러에 너그러워짐

비교 대상 데이터

- 본석님.csv (전부 Categorical Variable)
  - 'R_CL0000911', 'R_CL0000912', 'R_CS0000050', 'R_CS0000202', 'R_LC0017002', 'R_LC0017202', 'R_LC0321001', 'R_P13003300', 'R_P32001500', 'R_P32001700', 'R_P32002400', 'R_P44003900', 'R_P44003901', 'R_PE0000028', 'R_PE0000047', 'R_PE0000105', 'R_PH0000099', 'R_PH0000120', 'R_PH0000127', 'R_PHC000023', 'R_PS0000100', 'R_PS0000113', 'R_PS0000206', 'R_PS0001721', 'R_PS0001726', 'R_PS0001727', 'R_PS0001731', 'R_PS0001896', 'R_SC0000018', 'R_SC0000020', 'R_SC0000052'

- Remission.csv
  - "Remiss", "Cell", "Smear", "Infil", "Li", "Blast", "Temp"

데이터1: SAS and R (Entry 0.35, Stay 0.3)

Summary of Stepwise Selection							
Step	SAS		R		Score	Wald	Pr > ChiSq
	Entered	Removed	Entered	Removed	Chi-Square	Chi-Square	
1	R_P32002400		R_P32002400		288.2923		<.0001
2	R_CS0000202		R_CS0000202		111.812		<.0001
3	R_LC0017202		R_LC0017202		88.7185		<.0001
4	R_LC0321001		R_LC0321001		56.5524		<.0001
5	R_LC0017002		R_LC0017002		29.861		<.0001
6	R_CL0000912		R_CL0000912		27.863		<.0001
7	R_P44003901		R_P44003901		23.8204		<.0001
8	R_CS0000050		R_CS0000050		10.0662		0.0015
9	R_PS0000100		R_PS0000100		15.9769		0.0011
10	R_SC0000020		R_SC0000020		7.9237		0.019
11	R_P32001700		R_P32001700		5.9824		0.0502

데이터1: SAS and R (Entry 0.5, Stay 0.3)

Summary of Stepwise Selection							
Step	SAS		R		Score	Wald	Pr > ChiSq
	Entered	Removed	Entered	Removed	Chi-Square	Chi-Square	
1	R_P32002400		R_P32002400		288.2923		<.0001
2	R_CS0000202		R_CS0000202		111.812		<.0001
3	R_LC0017202		R_LC0017202		88.7185		<.0001
4	R_LC0321001		R_LC0321001		56.5524		<.0001
5	R_LC0017002		R_LC0017002		29.861		<.0001
6	R_CL0000912		R_CL0000912		27.863		<.0001
7	R_P44003901		R_P44003901		23.8204		<.0001
8	R_CS0000050		R_CS0000050		10.0662		0.0015
9	R_PS0000100		R_PS0000100		15.9769		0.0011
10	R_SC0000020		R_SC0000020		7.9237		0.019
11	R_P32001700		R_P32001700		5.9824		0.0502
12	R_PH0000099		R_PH0000099		3.2167		0.3594
13		R_PH0000099		R_PH0000099		3.213	0.3599

데이터1: SAS and R (Entry 0.5, Stay 0.6)

Step 17 까지 동일

Step 18에서 Perfect Multicollinearity 등장

waldtest.lm(before, after)에서 다음과 같은 에러가 발생했습니다:

there are aliased coefficients in the model

SAS는 해당 에러를 무시하고 쭉 진행

Multicollinearity 를 미리 제거하고 나면 문제가 안됨!

Linear Dependency

(Intercept)	-2.73546	R_PE00000281	-0.11125	R_PS00017261	0.142035
R_CL00009111	-0.74981	R_PE00000282	0.471866	R_PS00017262	0.203866
R_CL00009112	-0.72857	R_PE00000471	-0.36705	R_PS00017263	10.22674
R_CL00009113	-0.76254	R_PE00000472	-0.13717	R_PS00017271	-0.46198
R_CL00009114	-0.44652	R_PE00000473	NA	R_PS00017272	NA
R_CL00009121	0.949656	R_PE00001051	0.156519	R_PS00017273	NA
R_CL00009122	0.692156	R_PE00001052	-0.40227	R_PS00017311	-0.0548
R_CS00000501	0.178226	R_PH00000991	-0.10931	R_PS00017312	-0.00046
R_CS00002021	0.1522	R_PH00000992	-0.34576	R_PS00017313	-0.09443
R_LC00170021	0.336316	R_PH00000993	1.217138	R_PS00017314	-0.85308
R_LC00170022	0.4251	R_PH00001201	-0.18249	R_PS00018961	-0.04204
R_LC00172021	0.656496	R_PH00001202	NA	R_PS00018962	0.029339
R_LC00172022	0.708394	R_PH00001271	0.294053	R_PS00018963	-0.03383
R_LC03210011	0.205638	R_PH00001272	-0.82038	R_PS00018964	NA
R_LC03210012	0.493621	R_PHC0000231	NA	R_SC00000181	0.137793
R_P130033001	0.196744	R_PHC0000232	NA	R_SC00000182	0.113258
R_P130033002	1.19488	R_PHC0000233	NA	R_SC00000183	0.163698
R_P320015001	-0.12691	R_PS00001001	0.509184	R_SC00000201	0.088495
R_P320015002	-10.6937	R_PS00001002	-0.03899	R_SC00000202	0.139193
R_P320017001	0.307296	R_PS00001003	-0.0515	R_SC00000521	0.096724
R_P320017002	10.74321	R_PS00001131	NA	R_SC00000522	0.049711
R_P320024001	0.143059	R_PS00001132	NA	R_SC00000523	0.126816
R_P320024002	-0.28149	R_PS00001133	NA	R_SC00000524	0.149531
R_P440039001	0.628951	R_PS00002061	0.041194		
R_P440039002	0.67194	R_PS00002062	0.077193		
R_P440039003	0.690003	R_PS00002063	0.13087		
R_P440039004	0.511493	R_PS00002064	0.029441		
R_P440039011	NA	R_PS00017211	0.256233		
R_P440039012	-0.14078	R_PS00017212	-10.007		
R_P440039013	NA				

R_P440039011 =	R_P440039001 + R_P440039002
R_P440039013 =	R_P440039003 + R_P440039004 - R_P440039012
R_PE00000473 =	R_P440039001 + R_P440039002 + R_P440039003 + R_P440039004 - R_PE00000471 - R_PE00000472
R_PH00001202 =	R_PE00000281 + R_PE00000282 - R_PH00001201
R_PHC0000231 =	R_PE00000471
R_PHC0000232 =	R_PE00000472
R_PHC0000233 =	R_P440039001 + R_P440039002 + R_P440039003 + R_P440039004 - R_PE00000471 - R_PE00000472
R_PS00001131 =	R_P440039001 + R_P440039002
R_PS00001132 =	R_P440039012
R_PS00001133 =	R_P440039003 + R_P440039004 - R_P440039012
R_PS00017272 =	R_P130033001 - R_PS00017271
R_PS00017273 =	R_P130033002
R_PS00018964 =	R_PS00001003

```
# check linear dependency among variables
results <- fastDummies::dummy_cols(X,
                                   remove_first_dummy = TRUE,
                                   remove_selected_columns = TRUE)
lincomb = findLinearCombos(results)
lincomb_name = lapply(lincomb$linearCombos, function(x) colnames(results)[x])
print(lincomb_name)
```

데이터2: SAS Result (Entry 0.35, Stay 0.3)

Output 73.1.5: Summary of the Stepwise Selection

Summary of Stepwise Selection							
Step	Effect Entered	Effect Removed	DF	Number In	Score Chi-Square	Wald Chi-Square	Pr > ChiSq
1	li		1	1	7.9311		0.0049
2	temp		1	2	1.2591		0.2618
3	cell		1	3	1.4700		0.2254

링크:
https://documentation.sas.com/doc/en/statcdc/14.2/statug/statug_logistic_examples01.htm

	SAS		R		
Step	Entered	Removed	Entered	Removed	Equal
1	Li		Li		TRUE
2	Temp		Temp		TRUE
3	Cell		Cell		TRUE

Code Chunk (1) - Extractp
Estimated 된 모델에서 P value 뽑기

```
extractp <- function(pred, fitCurrent, step = 'fwd') {

  # Foward with Rao Score
  if (step == 'fwd'){
    before = fitCurrent
    after = fMaker(pred, fitCurrent, add=T) # Add
    pvalue = anova(before, after, test='Rao')[2,6] # Rao
    names(pvalue) = pred
    return(as.data.frame(pvalue))
  }

  # Backward with Wald Test
  else if (step == 'bwd')
    before = fitCurrent
    after = fMaker(pred, fitCurrent, add=F) # Remove
    pvalue <- lmtest::waldtest(before, after)[2,4] # Wald
    names(pvalue) = pred
    return(pvalue)
}
```

Code Chunk (2) - Stepfwd
전진선택법!!!

```
stepfwd <- function(fitCurrent, fullmodel, aEnter = 0.1, forcedOut = NULL) {
  predsInModel <- rownames(anova(fitCurrent))                                    #list of pre
  predsFull <- rownames(anova(fullmodel))                                        #list of pre
  predsNotInModel <- setdiff(predsFull, predsInModel)                            #list of pre
  pvalues <- unlist(sapply(predsNotInModel, function(x) as.numeric(extractp(x, fitCurrent, step = 'fwd'))))
  print(as.data.frame(round(pvalues,5)))
  cat('\n')
  if(length(pvalues)==0) return(fitCurrent)
  toAdd <- pvalues[which(pvalues==min(pvalues, na.rm = TRUE))]                                  #poss
  if(as.numeric(toAdd) <= aEnter) {
    cat("+++++ Add predictor", names(toAdd), "+++++", "\n")
    print(summary(fMaker(names(toAdd), fitCurrent))$coefficients, digits = 4)
    cat("\n")
    return(fMaker(names(toAdd), fitCurrent))                                     #updates an
  }
  return(fitCurrent)
}
```

Code Chunk (3) - Stepbwd
후진소거법!!!

```
stepbwd <- function(fitCurrent, fullmodel, aRemove = 0.1, forcedIn = NULL) {
  predsIncluded <- rownames(anova(fitCurrent))                                    #predictors
  predsIncluded <- predsIncluded[(predsIncluded != "NULL")]                       #removes "Nu
  predsIncluded <- setdiff(predsIncluded, intersect(predsIncluded, forcedIn))     #makes sure
  pvalues <- unlist(sapply(predsIncluded, function(x) as.numeric(extractp(x, fitCurrent, step = 'bwd'))))
  print(as.data.frame(round(pvalues,5)))
  cat('\n')
  if(length(pvalues)==0) return(fitCurrent)                                       #returns cur
  toRemove <- pvalues[which(pvalues==max(pvalues, na.rm = TRUE))]                 #se
  if(length(toRemove)==0) return(fitCurrent)
  if(as.numeric(toRemove) > aRemove){
    cat("----- Remove predictor", names(toRemove), "-----", "\n")
    print(summary(fMaker(names(toRemove), fitCurrent, add=FALSE))$coefficients, digits = 4)
    cat("\n")
    return(fMaker(names(toRemove), fitCurrent, add=FALSE))                        #returns an
  }
  return(fitCurrent)                                                            #else, returns
}
```

최종 단계적 선택법

```
else if(method=='stepwise'){
  # Step0 initiated
  fitBwd = stepfwd(fitBwd, fullmodel, forcedOut = forcedOut, aEnter = aEnter)
  step = 1
  num = length(attr(fullmodel$terms, 'dataClasses'))                    #
  while(-num < 0){
    print(paste0('Forward Selection Step: ', step))
    fitFwd = stepfwd(fitBwd, fullmodel, forcedOut = forcedOut, aEnter = aEnter)
    summary(fitFwd)
    if(identical(fitFwd, fitBwd) == T) {                                #
      cat("======== Final model ========", "\n")
      print(fitFwd$call)
      cat("Predictors forced in: ", forcedIn, "\n")
      cat("Predictors forced out: ", forcedOut, "\n", "\n")
      print(summary(fitFwd)$coefficients, digits = 4)
      cat("\n", "Alpha-to-enter = ", aEnter, ",  Alpha-to-remove = ", aRemove, "\n")
      return(invisible(fitFwd))
      break # no more addition
    }else {                                                             #
      print(paste0('Backward Selection Step: ', step))
      fitBwd = stepbwd(fitFwd, fullmodel, forcedIn = forcedIn, aRemove = aRemove)
      if(identical(fitFwd,fitBwd) != T){
        break # when there was a removal
      }
    }
    num = num + step
    step = step + 1
    }
}
```

Done