제주도 각 정류소의 퇴근시간 버스 승차인원 예측

2018580008 통계학과 문혜현

요약

제주도 버스의 효율적인 운행을 위해 퇴근시간 승차인원을 예측하는 모델을 만든다.

Ⅰ. 서론

1. 분석 목표

2. 데이터 소개

3. 데이터 전처리

4. 데이터 탐색

Ⅱ. 본론

1. 회귀분석

1.1. 회귀분석 개요

1.2. 선형모델(다중선형회귀분석)

1.3. 변수선택법(Ridge, Lasso, PCR)

1.4. 비선형 모델 (GAMs, Polynomial, Spline)

1.5. Tree 기반 모델(Regression Tree, Bagging, RandomForest)

1.6. 최종결과 및 데이터 분석

2. 분류분석

2.1. 분류분석 개요

2.2. 선형모델(로지스틱회귀분석)

2.3. 선형·이차판별분석(LDA, QDA)

2.4. 서포트벡터분류기

2.5. Tree 기반 모델(Classification Tree, Random Forest)

2.6. 최종결과 및 데이터 분석

Ⅲ. 결론

1. 최종 결과 요약

2. 활용방안

Ⅰ. 서론

1. 분석 목표

각 정류소의 제주도 퇴근시간 버스 승차인원에 대해 설명변수들이 어떠한 방식으로 영향을 주는지 분석합니다. 그리고 이를 바탕으로 각 정류소의 퇴근시간 버스 승차인원을 정확히 예측하는 모델을 세웁니다. 또한, 퇴근시간 버스 승차인원이 매우 많은 정류소들은 어떤 특징을 가지고 있는지 분석합니다. 이를 통해 미리 퇴근시간 승차인원이 많은 정류소들을 알아내어 퇴근시간 교통 혼잡함을 예방효과를 기대할 수 있습니다.

2. 데이터 소개

제주도 2019년 9월 1일부터 2019년 9월 30일에 제주도 내 모든 정류소에서의 승하차 인원을 기록한 데이터입니다.

어느 정류소에서 한 명의 승객이 승차 또는 하차했을 경우를 모두 기록하여 44만개의 관측치가 있습니다. 총 21개의 컬럼을 가지고 있으며 아래 표에 설명이 있습니다.

Column Name	Detail	Column Name	Detail
Id	해당 데이터에서의 고유한 ID	Latitude	해당 버스 정류장의 위도
Date	날짜	Longitude	해당 버스 정류장의 경도
Bus_route_id	노선ID	X~Y ride	X:00:00부터 X:59:59까지 승차한 인원 수(6시~12시까지 1시간 간격)
In_out	시내버스, 시외버스 구분	X~Y takeoff	X:00:00부터 X:59:59까지 하차한 인원 수(6시~12시까지 1시간 간격)
Station_code	해당 승하차 정류소의 ID	18~20 ride	18:00:00부터 19:59:59까지 승차한 인원 수
Station_name	해당 승하차 정류소의 이름		

3. 데이터 전처리

(1) 데이터 추출

분석의 편의성을 위하여 44만개의 데이터 중 일부를 추출하여 분석에 사용하였습니다. 2019년 9월 1일부터 2019년 9월 7일, 2019년 9월 15일부터 2019년 9월 21일 총 2주 동안의 데이터를 추출하였고 약 17만 8천개 입니다. 2019년 9월 12, 13, 14 일이 추석 연휴이기 때문에 2019년 9월 8일부터 2019년 9월 14일 대신 15일부터 21일을 추출했습니다. 연휴, 특별한 행사 등이 없는 일상생활에서의 버스 승차인원 예측을 다룰 것입니다.

(2) 컬럼 삭제

분석에 도움이 되지 않거나 필요 없는 컬럼들 id, station_name, latitude, longtitude, bus_route_id를 삭제했습니다.

Id는 관측치의 개수를 세는 의미 없는 컬럼입니다.

station_name은 다른 정류소이지만 비슷한 위치에 있으면 이름이 같은 경우가 있어 버스 정류장을 식별하는 데 좋은 변수가 아니며, station_code(버스정류장 식별번호)가 있기 때문에 필요 없는 변수입니다. latitude, longtitude도 마찬가지로 station_code(버스정류장 식별번호)가 위도, 경도에 포함된 정보를 가지고 있습니다. 정류소는 하나 당 위도와 경도 짝 하나가 할당되어 있기 때문입니다.

bus_rouote_id는 버스 노선이 퇴근시간 버스 승차인원에 큰 의미를 준다 생각하지 않았고, 이 역시 station_code와 비슷한 정보를 담고 있다고 생각했습니다.

(3) 컬럼 생성

아침 6시부터 오후 12시까지 매 1시간 간격마다 승차, 하차 인원이 기록되어 있습니다. 이 경우 설명변수의 개수가 너무 많고 각 컬럼마다 명확한 의미를 가지고 있지 않는 것으로 생각했습니다. 그래서 각 시간대에 맞는 특징인 새벽시간(6시~8시), 출근시간(8시~10시), 낮시간(10시~12시)을 생각했고, 기존의 시간대별 승차, 하차 인원의 변수들을 없애고 2시간 단위로 승차, 하차 인원을 합한 총 여섯 개의 변수들을 만들었습니다.

(4) 질적 변수 처리

시내버스와 시외 버스를 구분하는 bus_in 컬럼을 one-hot encoding 방식을 사용하여 더미화를 해주었습니다. 시내버스를 1로 두었습니다.

(5) 각 정류소 별로 데이터 그룹화

분석 목표는 퇴근시간 승차 인원을 하나하나 모두 예측하는 것이 아닌 각 정류소 별 퇴근시간 승차 인원을 예측하는 것입니다. 따라서 각 날짜마다 약 3000개의 정류소를 기준으로 각 설명변수의 승차, 하차 인원을 합했습니다.

(6) 분석을 위한 최종 데이터셋

```
> str(bus.ex.try2)
'data.frame':   33471 obs. of  8 variables:
 $ bus_in      : num  1 1 1 1 1 1 1 1 1 1 ...
 $ dawn_ride   : num  4 5 0 15 2 2 21 0 16 2 ...
 $ dawn_takeoff: num  0 1 0 2 2 1 0 0 2 1 ...
 $ work_ride   : num  2 13 4 16 2 5 11 0 34 4 ...
 $ work_takeoff: num  0 9 3 1 3 1 1 1 4 4 ...
 $ day_ride    : num  2 10 3 12 6 1 14 3 40 8 ...
 $ day_takeoff : num  0 5 4 0 2 4 3 0 13 6 ...
 $ y           : num  7 17 0 7 1 1 8 4 24 28 ...
>
```

Bus_in : 시내, 시외버스
Dawn_ride : 6~8시 승차인원
Dawn_takeoff :6~8시 하차인원
Work_ride : 8~10시 승차인원
Work_takeoff :8~10시 하차인원
Day_ride : 8~10시 승차인원
Day_takeoff : 8~10시 하차인원
Y : 18~20시 승차인원

4. 데이터 탐색

(1) 분석할 데이터의 크기는?

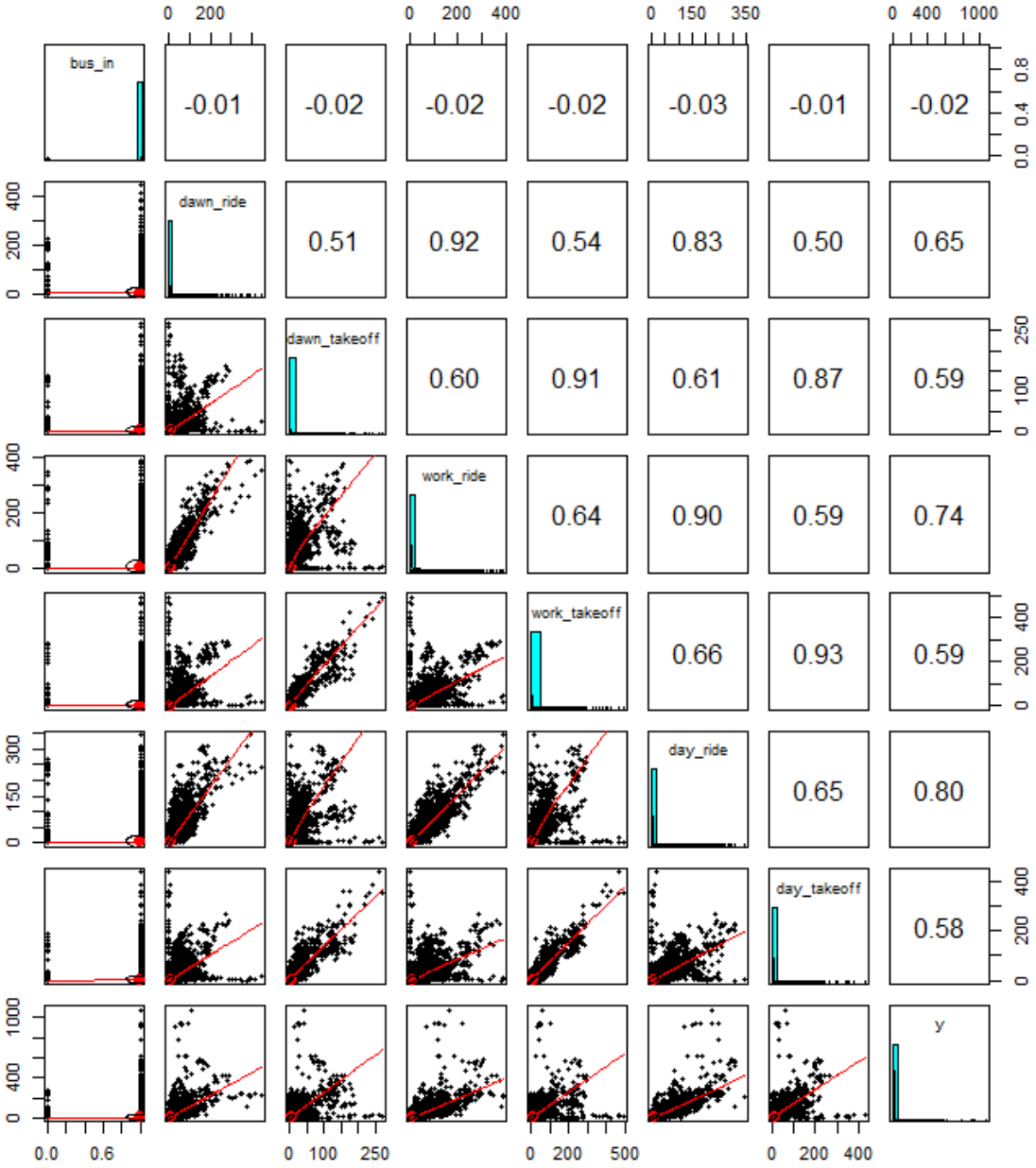
총 33471개의 관측치가 있고, 7개의 설명 변수가 있습니다.

(2) Missing value는 있는가?

Missing Value(결측치)는 없습니다.

(3) 데이터 시각화

아래 그래프는 X와 Y 각각의 분포, X와 Y 사이의 관계, X 서로의 관계, 상관계수 등을 시각화해 표현한 그래프입니다. 이 그래프를 통해 데이터를 더 깊이 탐색해 보겠습니다.



(4) Y(반응변수)의 정규성

Y는 정규성이 없습니다. Y에 대해 Log1p 함수를 적용해 데이터 변환을 통한 정규성을 찾아내려 하였지만 함수를 적용해도 정규성을 띄지 않았습니다.

실제로 Y는 3번째 분위수부터 3명 이상의 각 버스정류소의 퇴근시간 승차인원이 관측됩니다. 최대값은 1077로 대부분의 정류소가 퇴근시간 승차인원이 0이며 소수의 정류소들이 매우 높은 퇴근시간 버스 승차인원을 가지는 것으로 파악됩니다.

(5) X(설명변수) 사이의 관계

정류소를 기준으로 그룹화 했기 때문에 설명변수들 사이에 큰 연관성이 보입니다. 주로 dawn_ride, day_ride, work_ride의 탑승과 관련된 변수들이 서로 비례하고, dawn_takeoff, day_takeoff, work_takeoff의 하차와 관련된 변수들이 서로 비례합니다.

그리고 탑승과 관련된 변수들과 하차와 관련된 변수들은 반비례 관계에 있는 것을 그래프를 통해 확인할 수 있습니다.

(6) X(설명변수)와 Y(반응변수)의 관계

시외버스에 비해 시내버스가 퇴근시간 승차인원이 많은 것을 알 수 있습니다.

Dawn_takeoff(0.59), Work_takeoff(0.59), Day_takeoff(0.58)는 선형성을 보이기보다 반비례하는 것을 알 수 있습니다.

조금 더 자세히 들여다보기 위해 Work takeoff와 y plot을 살펴보도록 하겠습니다. 전체적으로는 반비례처럼 보여 하차인원이 적을수록 퇴근시간 승차인원이 많은 것처럼 보입니다. 하지만, 지나치게 하차인원과 승차인원이 많은 데이터(점)들을 제외하고 보면 양의 상관성을 가지며 비례함을 볼 수 있습니다.

다른 두 변수 Dawn takeoff, Day takeoff도 마찬가지의 양상을 띠고 있는 것을 확인할 수 있습니다.

Dawn_ride(0.65), Work_ride(0.74), Day_ride(0.8)는 y와 점점 더 큰 양의 연관성을 가지며 선형성을 볼 수 있습니다. 출근시간 승차인원이 많을수록 퇴근시간 승차인원이 많아지는 것을 볼 수 있습니다.

(7) 그래프를 통한 데이터 해석

새벽, 출근시간, 낮 시간 모두에서 승차와 하차 변수들과 퇴근시간 승차인원은 대체로 양의 상관성을 띄는 것을 볼 수 있습니다. 특히 퇴근시간 승차인원이 각 정류장에서 0~400 대 정도에서 이 현상을 확인할 수 있습니다.

예시를 들어 설명해 보자면, A 정류소는 출근시간에 승차인원이 10명이면, 하차인원도 10명이고, 퇴근시간 승차인원도 10명입니다. B 정류소는 출근시간에 승차인원이 3명이면, 하차인원도 3명이고, 퇴근시간 승차인원도 3명입니다. A, B 모두 그 만큼의 유동인구가 다니는 길목에 위치해 있기 때문에 특정한 이유가 없는 한 그 정류소 근처의 유동인구만큼 사람들이 각 시간대마다 이용할

것입니다.

승차와 관련된 변수들에서는 출근시간대 승하차인원 대비 퇴근시간 승차 인원이 매우 적거나 매우 큰 값들에 대해 다음과 같은 패턴을 알 수 있습니다. 새벽, 출근시간, 낮으로 시간이 변함에 따라 버스를 이용하는 승객들이 많아져 시간에 따라 점차 수치들이 높아집니다. 특히, 승차 인원 대비 퇴근시간 승차가 많은 정류장들이 눈에 띄게 시간에 따라 오른쪽으로 이동하는 것이 보입니다. 이는 그 정류소들이 낮이 되어가면서 유동인구가 많아지는 장소에 있다는 것을 보여주는 것으로 생각합니다.

하차와 관련된 변수들에서는 출근시간대 승하차인원 대비 퇴근시간 승차 인원이 매우 적거나 매우 큰 값들에 대해 다음과 같은 패턴을 알 수 있습니다. 새벽, 출근시간, 낮에 하차인원이 적은 곳은 퇴근시간 승차 인원이 많고, 새벽, 출근시간, 낮에 하차인원이 많은 곳은 퇴근시간 승차 인원이 적습니다. 이는 보통 한 장소에 반대편 방향으로 가는 것을 감안해 정류소가 2개 있는 것을 생각해야 합니다. 따라서, 이 정류소들은 한 장소에 짝으로 있는 정류소들일 것입니다.

이 두 가지 해석을 통해 퇴근시간 승차인원이 다른 곳에 비해 많은 곳은, 즉 교통의 혼잡을 발생시키는 정류소들은 주로 회사가 몰려 있는 도심일 것입니다.(실제로 이 정류소들은 제주도의 큰 도시에 많이 분포해 있습니다. 50p 지도 참고) 그리고 이것이 전체적으로 승하차인원과 퇴근시간 승하차인원이 양의 상관성을 갖는 패턴에서 몇몇 정류소가 벗어나는 특정한 이유일 것입니다.

Ⅱ. 본론

1. 회귀분석

1.1. 회귀분석 개요

(1) 분석 목적

데이터 분석과 예측을 하면서 주목할 점은 두 가지입니다. 첫 번째는 전체적인 퇴근시간 승차인원에 영향을 주는 새벽, 출근시간, 낮의 승하차 설명변수의 특징입니다. 두 번째는 새벽, 출근시간, 낮 승하차인원 대비 퇴근시간 승차 인원이 매우 큰 값들입니다. 왜냐하면 이 부분들이 전체적인 퇴근시간 승차인원의 경향과 벗어나는 패턴을 가지고 있어 이에 유의하며 설명변수들의 특징을 해석해야 합니다.

(2) 선택한 모델 기법

크게 세 가지 측면으로 나누어 모델링을 할 것입니다. 설명변수와 반응변수 간의 선형성이 나타났으므로 선형모델에 적합할 것입니다. 두 번째로, 하차 관련 변수들이 비선형성을 띄었기 때문에 비선형모델에 적합할 것입니다. 세 번째로 비선형과 선형도 아닌 일반적인 분포(모델)이라 생각한다면 Tree 기반 모델이 잘 적합될 것입니다.

(3) Train, validation set 분할

회귀분석에 앞서 데이터를 train, validation set으로 나누어 줍니다. createDataPartition 함수를 사용하여 퇴근시간 승하차인원이 매우 작거나 큰 수치들이 train, validation set에 각각 몰리는 것을 방지했습니다. 각 train, validation set에 y 수치들을 일정한 비율로 넣어주었습니다.

```
> set.seed(1)
> idx<-createDataPartition(bus.ex.try2$y, p=0.7, list=FALSE)
> bus.ex.train<-bus.ex.try2[idx,]
> bus.ex.test<-bus.ex.try2[-idx,]
> summary(bus.ex.train$y)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
   0.000    0.000    0.000    6.581    3.000 1077.000
> summary(bus.ex.test$y)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   0.000   0.000   7.175   3.000 946.000
```

1.2. 다중선형회귀모델

(i) 선정 이유

데이터 탐색을 통해 승차와 관련된 변수들과 퇴근시간 승차인원이 강한 선형성을 갖는 것을 볼

수 있었습니다. 또한, 하차와 관련된 변수들 역시 선형성을 일부분 가지는 모양을 포착할 수 있었습니다. 따라서 가장 단순한 선형모델인 다중선형회귀모델을 적합해 반응변수에 대한 각 설명 변수의 영향을 해석하고 다중선형회귀모델의 적합성을 검정하려 합니다.

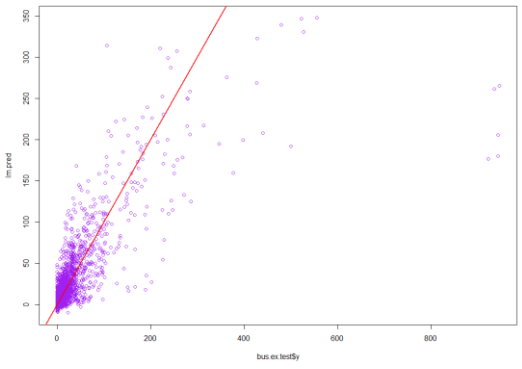
(ii) 적합 및 예측

Bus_in을 제외한 모든 변수들의 계수추정치가 유의함을 확인할 수 있습니다. Anova를 이용해 F 검정을 했을 때, 모든 변수가 모두 유의하며 반응변수인 퇴근시간 승차인원과 관련이 있음을 알 수 있습니다. 설명력 Adjusted R^2 =0.6986이며 적합한 다중선형회귀를 이용해 예측을 한 결과 rmse=20.84169 가 나왔습니다.

```
> set.seed(1)
> lm.fit<-lm(y~., data=bus.ex.train)
> summary(lm.fit)

Call:
lm(formula = y ~ ., data = bus.ex.train)

Residuals:
    Min      1Q  Median      3Q     Max
-141.02   -1.30    0.49    1.34  827.56

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -2.04463    0.73350  -2.788  0.00532 **
bus_in        0.96993    0.73851   1.313  0.18908
dawn_ride    -0.19566    0.01391 -14.070  < 2e-16 ***
dawn_takeoff  0.82022    0.02405  34.098  < 2e-16 ***
work_ride     0.26023    0.01489  17.480  < 2e-16 ***
work_takeoff -0.19190    0.01738 -11.042  < 2e-16 ***
day_ride      0.95451    0.01280  74.578  < 2e-16 ***
day_takeoff  -0.04561    0.01809  -2.521  0.01171 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.82 on 23423 degrees of freedom
Multiple R-squared:  0.6987,    Adjusted R-squared:  0.6986
F-statistic:  7759 on 7 and 23423 DF,  p-value: < 2.2e-16
```

```
> anova(lm.fit)
Analysis of Variance Table

Response: y
               Df  Sum Sq Mean Sq    F value    Pr(>F)
bus_in          1    9828    9828    44.7474 2.292e-11 ***
dawn_ride       1 7868713 7868713 35824.9387 < 2.2e-16 ***
dawn_takeoff    1 1549654 1549654  7055.3152 < 2.2e-16 ***
work_ride       1 1189359 1189359  5414.9536 < 2.2e-16 ***
work_takeoff    1    6243    6243    28.4239 9.836e-08 ***
day_ride        1 1303875 1303875  5936.3240 < 2.2e-16 ***
day_takeoff     1    1396    1396     6.3552   0.01171 *
Residuals   23423 5144709     220
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


> lm.pred<-predict(lm.fit, newdata=bus.ex.test )
> rmse(bus.ex.test$y, lm.pred)
[1] 20.84169
>
> ##예측 정확성 판단하기
> plot(bus.ex.test$y, lm.pred, col=c(1,2))
```

적합한 값과 실제 관측값을 그래프를 통해 보면 퇴근시간 승차 인원이 200 이하인 값들에 대해서는 어느 정도 잘 예측함을 볼 수 있습니다. 하지만, 퇴근시간 승차인원이 큰 값들에 대해 과소추정하는 경향이 있음을 알 수 있습니다.



(iii) 성능 개선

다중선형회귀모델의 성능을 좀 더 향상시키기 위해 변수선택법을 활용해 변수를 선택하여 다중공선성을 완화하려 합니다.

앞서 데이터 탐색에서 승차관련 변수들과 하차관련 변수들끼리 상관관계가 큰 것을 알 수 있었습니다. 이는 적합한 다중선형회귀모델에서 다중공선성 문제를 일으켜 적합을 방해할 수 있습니다. Vif 함수를 통해 10 이상인 work_ride와 work_takeoff 가 다중공선성을 일으킨다고 볼 수 있습니다.

```
> vif(lm.fit)
      bus_in    dawn_ride dawn_takeoff    work_ride
    1.002658     6.501675     5.974607    11.457603
work_takeoff     day_ride  day_takeoff
   11.185277     6.055979     8.167574
```

다중공선성이 모형의 예측력을 방해하므로 다중공선성을 해결하면 좀 더 데이터들을 잘 예측할 수 있을 것입니다. 또한, 회귀계수추정치들이 반응변수와 각 설명변수들 간의 관계를 더 명확히 잘 나타낼 것으로 예상했습니다.

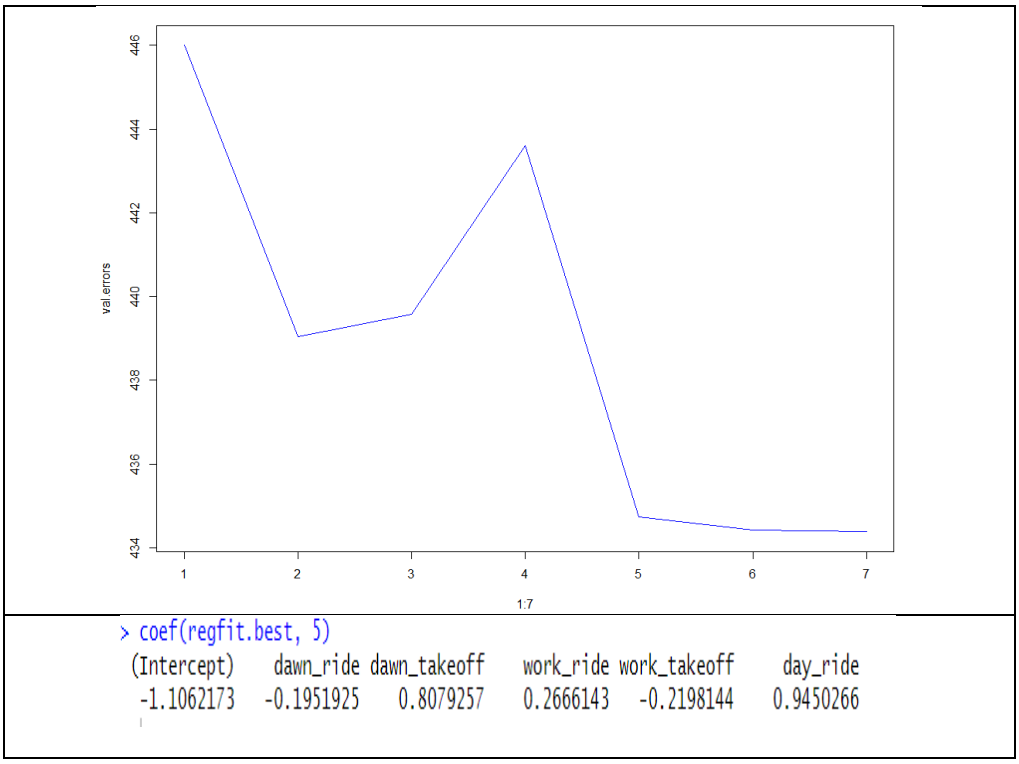
따라서 다중공선성 대한 해결책으로 최상의 부분집합 선택법을 이용하겠습니다. 어떤 변수 조합이 가장 성능이 좋으며, 이에 따라 다중공선성 문제도 해결이 되는지 확인해 보도록 하겠습니다.

(iv) 성능 개선-최상의 부분집합 선택법

① Train set에 대해서 최상의 부분집합선택법 시행

①-1. 각 변수개수에 따른 검증오차를 사용하여 변수를 선택하였습니다.

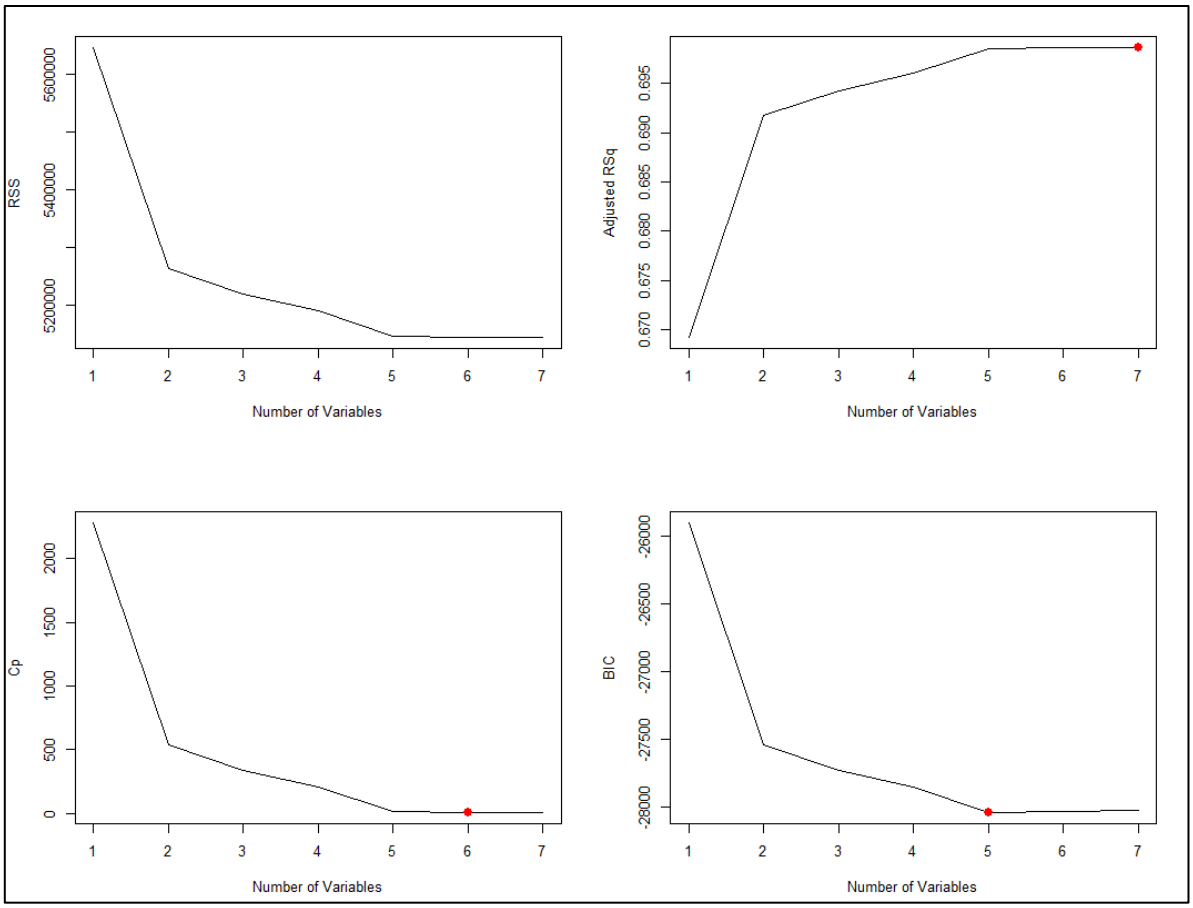
변수가 7개 모두 사용되었을 때 검증오차가 가장 낮았지만 변수선택에 의미를 두었기 때문에 5개를 선정하였습니다. Dawn_ride, dawn_takeoff, work_ride, work_takeoff, day_ride 총 5개의 변수가 선택이 되었습니다.




```
> coef(regfit.best, 5)
(Intercept)    dawn_ride dawn_takeoff    work_ride work_takeoff     day_ride
 -1.1062173  -0.1951925    0.8079257    0.2666143  -0.2198144    0.9450266
```

①-2. RSS, Adj R^2, Cp, BIC 를 사용하여 변수를 선택하였습니다.

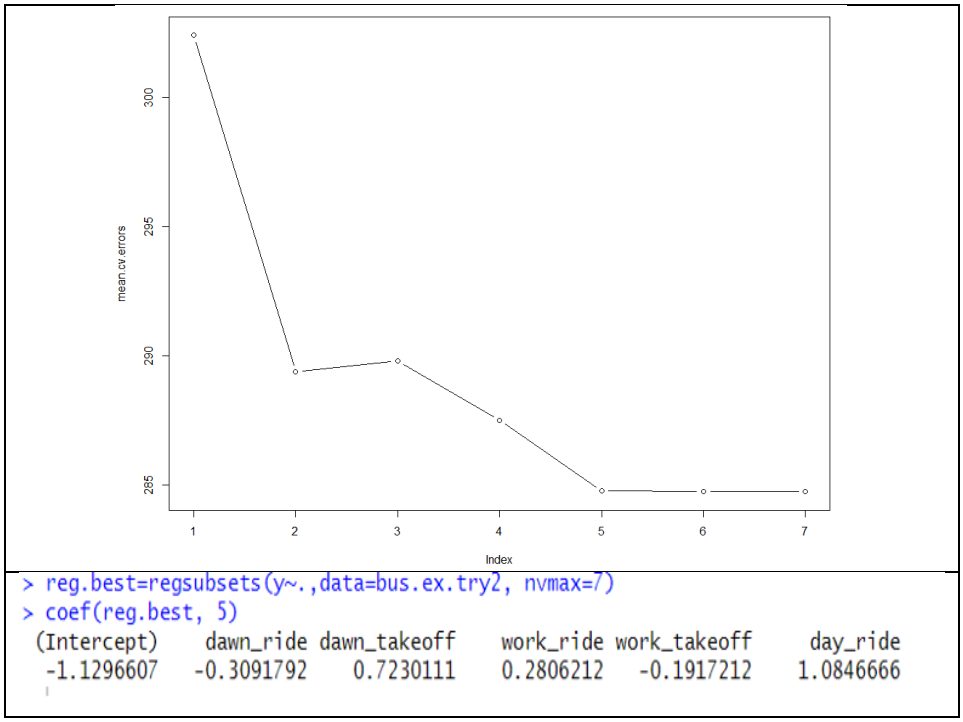
변수가 늘어날수록 오차가 점점 줄어드는 것을 네 개의 plot을 통해 볼 수 있습니다. 5개~6개 이후부터 RSS Adj R^2, Cp가 많이 낮아지며 BIC도 많이 높아지는 것을 볼 수 있습니다. 따라서 변수 5개를 선택하고, 이는 앞에서 검증오차를 통해 선택한 변수의 개수와 똑같음을 알 수 있습니다.



② 전체 data set에 대해서 최상의 부분집합선택법 시행

전체 data set에 대해서 교차검증오차를 비교해 본 결과 변수개수를 5개 선택했을 때 오차가 현저히 낮아지는 것을 확인 하 수 있습니다.

전체 data set에서의 선택되는 변수 5개는 dawn_ride, dawn_takeoff, work_ride, work_takeoff, day_ride 입니다.




```
> reg.best=regsubsets(y~.,data=bus.ex.try2, nvmax=7)
> coef(reg.best, 5)
 (Intercept)    dawn_ride dawn_takeoff    work_ride work_takeoff     day_ride
  -1.1296607   -0.3091792    0.7230111    0.2806212   -0.1917212    1.0846666
```

③ ①과 ② 비교

두 data set에 대해서 최상의 부분집합선택법을 시행했을 때 dawn_ride, dawn_takeoff, work_ride, work_takeoff, day_ride 5개의 변수가 모두 같게 나오는 것을 확인 할 수 있습니다.

또한, 계수추정치를 비교해 보면 부호는 모두 같지만 수치가 차이가 남을 확인할 수 있는데 눈에 띄게 크게 차이나는 것은 없습니다. 따라서 최상의 부분집합선택법을 통해 구한 5개의 변수를 이용해 다중선형회귀를 한 번 더 적합해보고, 모형이 유의해졌는지, 예측력이 올라갔는지, 다중공선성이 해결이 되었는지 확인해 보도록 하겠습니다.

Train set에 대해서 부분집합선택법 시행	전체 data set에 대해서 부분집합선택법 시행
> coef(regfit.best, 5) (Intercept) dawn_ride dawn_takeoff work_ride work_takeoff day_ride -1.1062173 -0.1951925 0.8079257 0.2666143 -0.2198144 0.9450266	> coef(reg.best, 5) (Intercept) dawn_ride dawn_takeoff work_ride work_takeoff day_ride -1.1296607 -0.3091792 0.7230111 0.2806212 -0.1917212 1.0846666 >

(v) 성능 개선 모델에 대한 적합 및 예측

모든 설명변수들의 계수추정치가 유의함을 확인할 수 있습니다. Anova를 이용해 F 검정을 했을 때, 모든 설명변수가 모두 유의하며 반응변수인 퇴근시간 승차인원과 관련이 있음을 알 수 있습니다. 설명력 Adjusted R^2 =0.6985이며 적합한 다중선형회귀를 이용해 예측을 한 결과 rmse=20.8505이 나왔습니다. 앞서 시행했던 다중선형회귀와 설명력이 비슷하며 rmse는 약 0.01 만큼 떨어졌습니다. 또한, 계수추정치도 부호가 같으며 비슷한 수치임을 확인할 수 있습니다.

다중공선성을 vif를 통해 확인한 결과 완화된 것을 볼 수 있지만 아직 남아있는 것을 알 수 있습니다.

또한, 잔차분석 표를 비교했을 때 눈에 띄게 나아진 점을 찾아볼 수 없으므로 모형의 개선이 보이지 않습니다.

결과적으로 최상의 부분집합 선택을 통해 적합한 다중선형회귀모델은 성능이 크게 나아지지 않은 것으로 결론을 내릴 수 있습니다.

```
> lm.fit2<-lm(y~dawn_ride+dawn_takeoff+work_ride+work_takeoff+day_ride, data=bus.ex.train)
> summary(lm.fit2)

Call:
lm(formula = y ~ dawn_ride + dawn_takeoff + work_ride + work_takeoff +
    day_ride, data = bus.ex.train)

Residuals:
    Min      1Q  Median      3Q     Max
-140.08   -1.30    0.52    1.33  828.13

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -1.10622    0.10378  -10.66   <2e-16 ***
dawn_ride    -0.19519    0.01390  -14.04   <2e-16 ***
dawn_takeoff  0.80793    0.02360   34.24   <2e-16 ***
work_ride     0.26661    0.01465   18.19   <2e-16 ***
work_takeoff -0.21981    0.01330  -16.53   <2e-16 ***
day_ride      0.94503    0.01227   77.03   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.82 on 23425 degrees of freedom
Multiple R-squared:  0.6986,    Adjusted R-squared:  0.6985
F-statistic: 1.086e+04 on 5 and 23425 DF,  p-value: < 2.2e-16
```

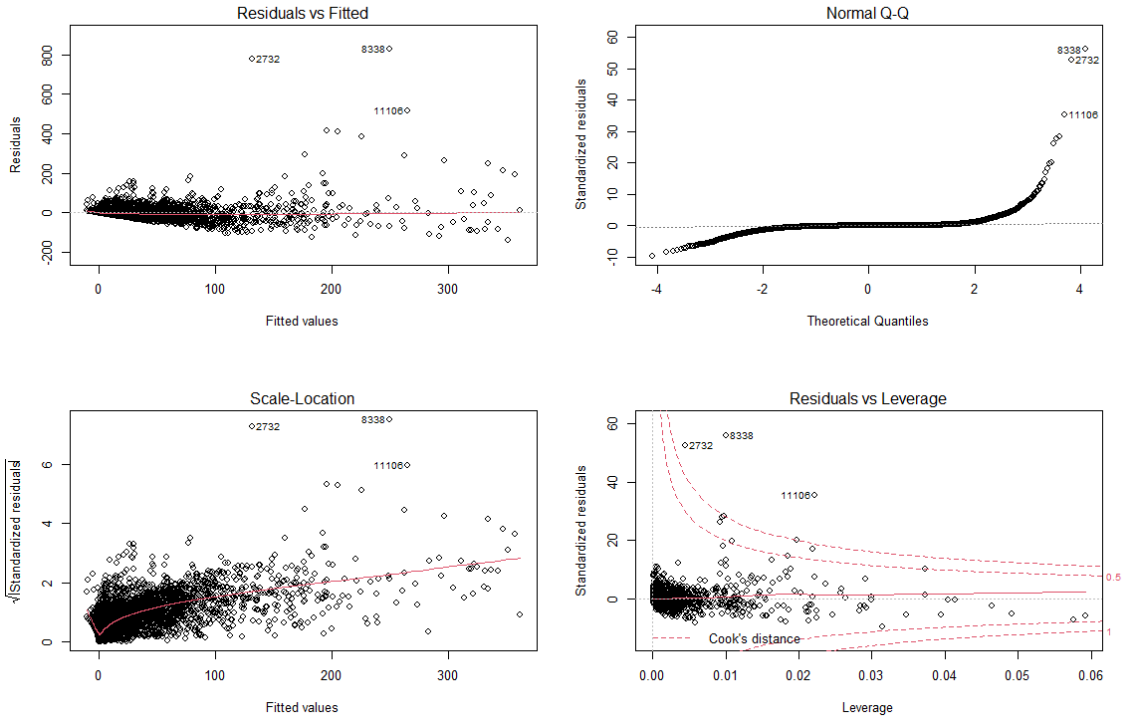
```
> anova(lm.fit2)
Analysis of Variance Table

Response: y
              Df  Sum Sq Mean Sq  F value    Pr(>F)
dawn_ride      1 7875210 7875210 35845.487 < 2.2e-16 ***
dawn_takeoff   1 1551953 1551953  7064.003 < 2.2e-16 ***
work_ride      1 1190367 1190367  5418.179 < 2.2e-16 ***
work_takeoff   1    6254    6254    28.465 9.63e-08 ***
day_ride       1 1303549 1303549  5933.346 < 2.2e-16 ***
Residuals  23425 5146444     220
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

```
> lm.pred2<-predict(lm.fit2, newdata=bus.ex.test)
> rmse(bus.ex.test$y, lm.pred2)
[1] 20.8505
```

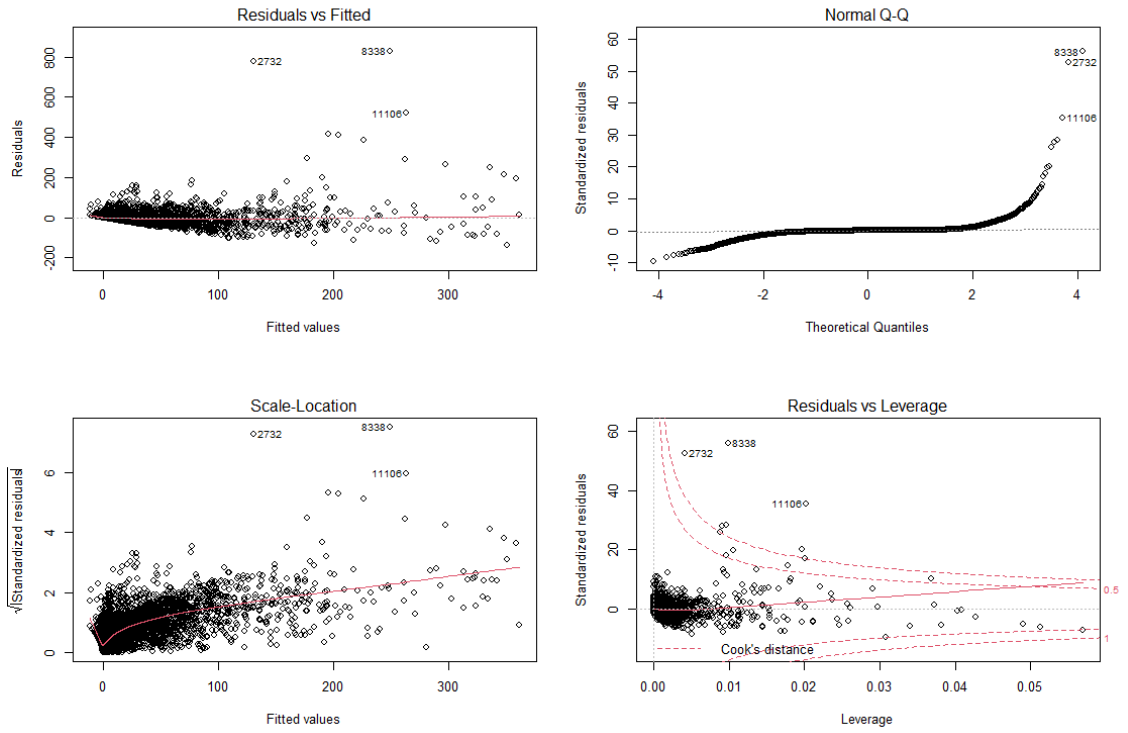
```
> vif(lm.fit2)
  dawn_ride dawn_takeoff   work_ride work_takeoff    day_ride
   6.497927    5.749060   11.098999    6.545982    5.563281
```

<잔차분석 그래프>

<모든 설명변수를 넣은 다중선형회귀모델에 대한 잔차 그래프>



<최상의 부분선택법을 이용한 다중선형회귀모델에 대한 잔차 그래프>



(vi) 결론

결론적으로 다음 이유에서 다중선형회귀모델은 제주도 버스 퇴근시간 승차인원 예측 데이터에 적합한 모델이 아닙니다.

1. 반응변수가 정규성을 크게 따르지 않습니다.
2. 데이터 탐색을 통해 설명변수와 반응변수 사이의 관계가 선형성도 있었지만 비선형성도 있었기 때문입니다.
3. 이상치의 존재들이 많았고, 이에 따라 레버리지가 높은(영향력이 큰) 관측치들도 많을 것임을 알 수 있습니다. 그리고 이는 앞에 나왔던 잔차 분석 그래프를 통해 확인 할 수 있습니다.
4. 변수선택법으로 다중공선성을 완화하였으나 다중공선성 문제가 사라지지는 않았습니다.
5. 등분산과 공분산성이라는 오차 가정이 상당히 맞지 않는 것을 잔차 분석 그래프를 통해 볼 수 있습니다.

결론적으로, 다중선형회귀모델은 예측 면에서 출근시간 승차인원이 큰 값에서는 과소추정을 하는 경향이 있었고, 해석 면에서는 다중선형회귀의 기본 가정들이 맞지 않는 부분들이 있었기 때문에 모형이 유의하지 않을 것입니다. 따라서 계수추정치가 반응변수와 설명변수 사이의 관계를 올바르게 표현하는지를 의심해 보아야 합니다.

1.3.변수선택법

선형모델이면서 좀 더 유연하게 데이터를 예측하기 위해 최소제곱법이 아닌 계수추정치들을 제한하고 규칙화하는 기법을 사용하여 선형모델을 적합해봅니다. 능형회귀, Lasso, 그리고 주성분회귀를 사용할 것입니다.

모델적합에 앞서 데이터를 표준화할 것입니다. 사용하려는 모델들의 계수추정방법은 각 설명변수들의 단위가 다를 경우 계수추정치가 특정 설명변수의 큰 단위에 영향을 받아 올바르게 나오지 않을 수 있기 때문입니다. 표준화는 전체 데이터에서 해 준 뒤, 이전에 만들어 두었던 train, validation set과 동일하게 다시 분할해 줍니다.

```
> bus.ex.scale<-data.frame(bus_in=bus.ex.try$bus_in,scale(bus.ex.try2[,2:8]),y=bus.ex.try2$y)
> bus.scale.train<-bus.ex.scale[idx,]
> bus.scale.test<-bus.ex.scale[-idx,]
```

(1) Ridge Regression(능형회귀)

(i) 선정 이유

최소제곱추정방식 대신 능형회귀를 이용하는 이유는 편향-분산 절충 방식 때문입니다.

능형회귀의 계수추정방식은 최소제곱방식에 수축페널티항을 부여하는 것입니다. 수축페널티를 조절하는 λ가 0에 가까울수록 최소제곱추정에 가깝고, λ가 증가할수록 패널티가 커져 능형계수 추정치들이 수축합니다.

따라서 λ가 증가하면 능형회귀의 적합의 유연성이 감소하게 되어 분산은 감소하지만 편향은 증가합니다. 그렇기에 최소제곱 추정치가 높은 분산을 가지는 상황에서 가장 잘 동작합니다.

제주도 버스 승차인원 예측 데이터의 전체 설명변수를 모두 넣은 다중선형회귀에서 bus_in의 계수추정치의 분산이 큰 것 이외에 다른 설명변수들의 계수추정치의 분산은 크지 않았습니다. 하지만, 능형회귀를 통해 편향-분산 절충으로 더 좋은 모델을 구할 수 있다는 가능성이 있기 때문에 능형회귀를 이용하였습니다.

(ii) 적합 및 예측

교차검증 방식을 사용해 수축페널티를 조절하는 λ의 적절한 값을 구했고, $\lambda \approx 2.2084$ 가 나왔습니다. 이 값을 사용해 Train set 을 능형회귀에 적합하였고, validation set 에 대한 예측값을 구했습니다. 이 때 RMSE=21.43479 가 나왔습니다.

앞서 말했듯이, 퇴근시간 버스승차인원의 다중선형회귀모델에서 회귀계수추정치의 분산이 작은 경우였습니다. 따라서, 편향-절충 방식이 효과적으로 작동하지 않았고, 오히려 회귀계수추정치를 수축함으로서 예측력이 떨어졌음을 볼 수 있습니다.

```
> set.seed(1)
> cv.out=cv.glmnet(train.x, train.y, alpha=0, nfolds=5)
> bestlam=cv.out$lambda.min
> bestlam
[1] 2.208381
>
> ridge.fit=glmnet(train.x, train.y, alpha=0, lambda=bestlam)
> ridge.pred<-predict(ridge.fit, s=bestlam, newx=test.x)
>
> rmse(test.y, ridge.pred)
[1] 21.43479
```

```
> ridge.fit$beta
7 x 1 sparse Matrix of class "dgCMatrix"
                      s0
bus_in        0.171100438
dawn_ride    -0.007805311
dawn_takeoff  0.508250449
work_ride     0.249045763
work_takeoff -0.032585605
day_ride      0.710449412
day_takeoff   0.037508905
```

(2) Lasso

(i) 선정 이유

Ridge regression에서 모든 설명변수를 사용했다면, Lasso는 ℓ_2패널티 대신 ℓ_1패널티를 사용하여 λ 가 충분히 클 경우 변수 선택을 수행합니다. 따라서 해석이 더 용이하며 마찬가지로 편향-분산 절충방식을 사용합니다. 일반적으로 비교적 적은 수의 설명변수가 상당히 큰 계수를 가지고 나머지 변수들은 계수가 아주 작거나 0인 설정에서 성능이 더 낫습니다.

제주도 버스 승차인원 예측 데이터의 전체 설명변수를 모두 넣은 다중선형회귀에서 설명변수 7개 중 3개의 계수추정치 절댓값이 0.8 이상을 가지며 나머지 4개의 절대값이 0.01~0.2 사이입니다. 따라서, Lasso를 통한 변수선택을 통해 더 용이한 해석과 더 정확한 예측을 기대해 Lasso 방식을 이용했습니다.

(ii) 적합 및 예측

교차검증 방식을 사용해 조율 파라미터 λ의 적절한 값을 구했고, $\lambda \approx 0.0188$ 이 나왔습니다. 이 값을 사용해 Train set 을 Lasso 모델에 적합하였고, validation set 에 대한 예측값을 구했습니다. 이 때 RMSE=21.43479 가 나왔습니다.

λ가 매우 작기 때문에 회귀계수추정치가 0 이 되는 변수선택이 진행되지 않았고, 따라서 결과는 Ridge Regression model 과 거의 비슷합니다.

```
> set.seed(1)
> cv.out=cv.glmnet(train.x, train.y, alpha=1, nfolds=3)
> bestlam=cv.out$lambda.min
> bestlam
[1] 0.01876579
>
> lasso.fit=glmnet(train.x, train.y, alpha=1, lambda=bestl
> lasso.pred<-predict(ridge.fit, s=bestlam, newx=test.x)
> rmse(test.y, lasso.pred)
[1] 21.43479
```

```
> lasso.fit$beta
7 x 1 sparse Matrix of class "dgCMatri
                      s0
bus_in         0.78594197
dawn_ride     -0.17878607
dawn_takeoff   0.79971280
work_ride      0.24750182
work_takeoff  -0.18152707
day_ride       0.95080597
day_takeoff   -0.04075688
```

(3) PCR(주성분회귀)

(i) 선정 이유

설명변수들을 변화한 다음에 변환된 변수들을 사용해 최소제곱모델을 적합하는 차원축소기법을 사용합니다. 적은 수의 주성분들로 데이터 내 대부분의 변동과 반응변수와의 상관관계를 설명할 수 있다는 가정을 합니다. 결과적으로 이 가정이 성립하면 모든 설명변수를 사용하지 않고 추정함으로서 과적합을 줄일 수 있습니다.

따라서 PCR은 처음 몇 개의 주성분으로 설명변수들의 변동 대부분과 반응변수와의 상관관계를 얻을 수 있는 경우에 좋은 결과를 내는 경향이 있습니다.

제주도 버스 승차인원 예측 데이터가 이 가정이 성립하는지, 그리고 성립한다면 과적합을 방지할 수 있는 좋은 모델을 만들 수 있는지 알아보기 위해 PCR을 사용했습니다.

(PLS는 PCR과 비슷한 성능을 가지기 때문에 PLS는 굳이 하지 않았습니다.)

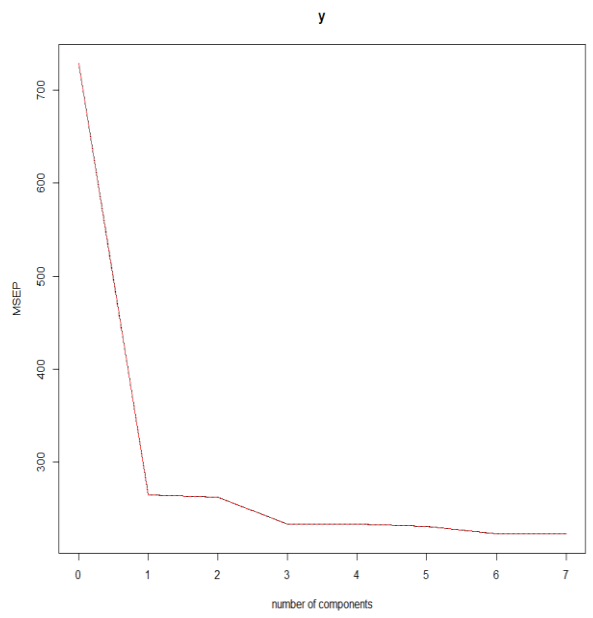
(ii) 적합 및 예측

교차검증 방식을 사용해 MSEP 가 최소가 되는 주성분 개수를 구하였고, MSEP 많이 낮아져 안정화가 되는 4 개를 주성분개수로 정하였습니다. 이 값을 사용해 Train set 을 주성분회귀모델에 적합하였고, validation set 에 대한 예측값을 구했습니다. 이 때 RMSE= 21.05421 가 나왔습니다.

처음 몇 개의 주성분으로 MSEP 가 급격히 줄어듦을 통해 적은 주성분으로 제주도 퇴근시간 승차인원에 대한 정보를 담아낼 수 있음을 알 수 있습니다.

선택한 주성분 개수 4개에 대해 다중선형회귀를 적합하였습니다. 그 때, 모든 계수가 유의하며 Adjusted R^2이 0.6822 임을 알 수 있습니다.

<주성분 개수에 따른 MSEP>



<주성분 개수를 다중선형회귀 적합>

```
> summary(pc.lm)

Call:
lm(formula = y ~ ., data = new.data)

Residuals:
    Min      1Q  Median      3Q     Max
-140.67   -1.15    0.58    1.23  844.16

Coefficients:
            Estimate Std. Error  t value Pr(>|t|)
(Intercept)  6.581153   0.099415   66.199  < 2e-16 ***
`Comp 1`    -0.571401   0.002638 -216.636  < 2e-16 ***
`Comp 2`     0.109282   0.005837   18.724  < 2e-16 ***
`Comp 3`    -0.734192   0.013411  -54.746  < 2e-16 ***
`Comp 4`    -0.093422   0.018976   -4.923 8.57e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 15.22 on 23426 degrees of freedom
Multiple R-squared:  0.6823,    Adjusted R-squared:  0.6822
F-statistic: 1.258e+04 on 4 and 23426 DF,  p-value: < 2.2e-16
```

(iii) 주성분에 대한 해석

① 주성분들의 상관계수를 통해 상관계수들이 모두 거의 0에 가깝고, 다중공선성이 해결된 것을 볼 수 있습니다.

② 4개의 주성분 개수에 대해서 좀 더 알아보도록 하겠습니다.

Comp1은 dawn_ride, work_ride, day_ride 로 주로 승차인원의 의미를 갖는 변수로 해석할 수 있습니다.

Comp2는 Work_takeoff, day_takeoff 로 주로 하차인원의 의미를 갖는 변수로 해석할 수 있습니다.

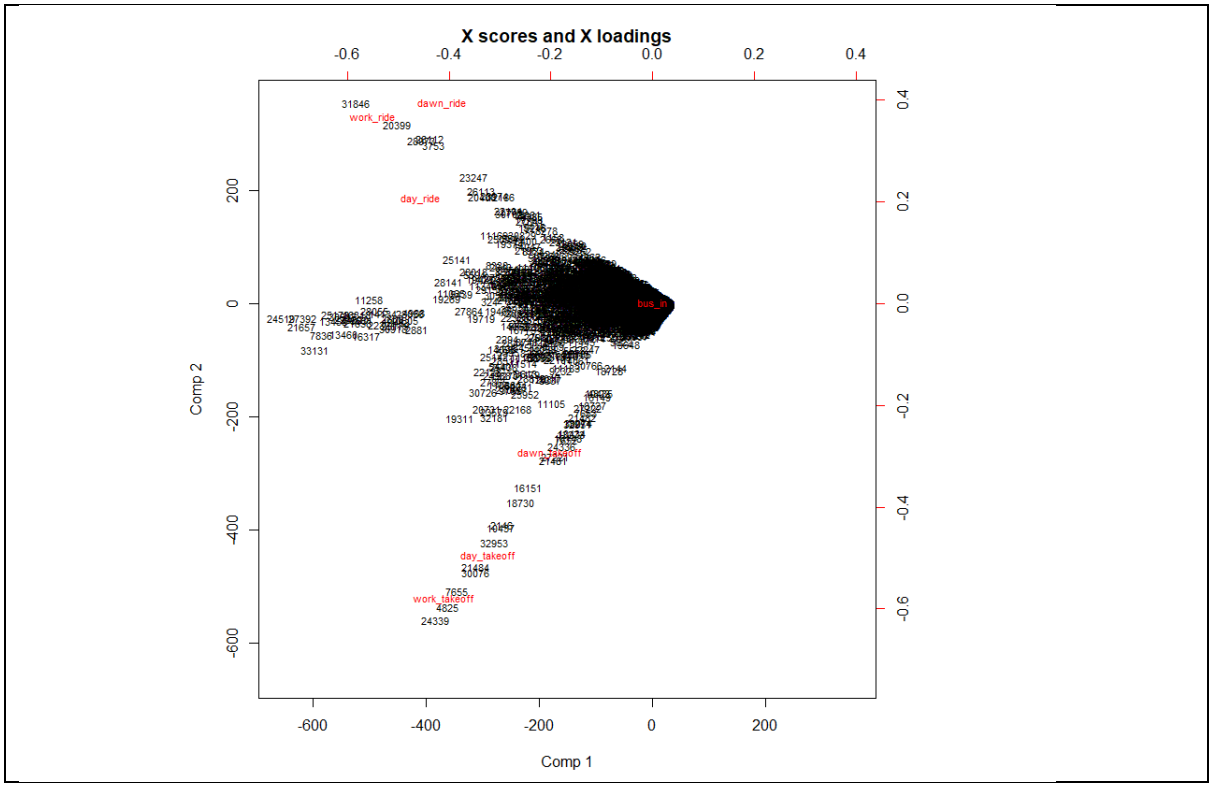
Comp3은 day_takeoff 로 낮시간 하차인원의 의미를 갖는 변수로 해석할 수 있습니다.

Comp4는 dawn_ride, work_ride 로 새벽, 낮 시간 승차인원의 의미를 갖는 변수로 해석할 수 있습니다.

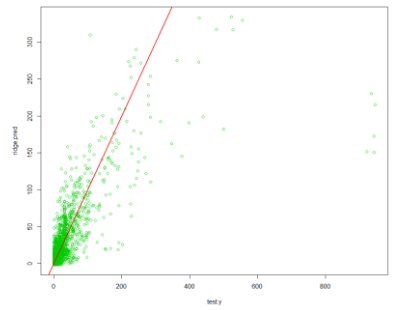
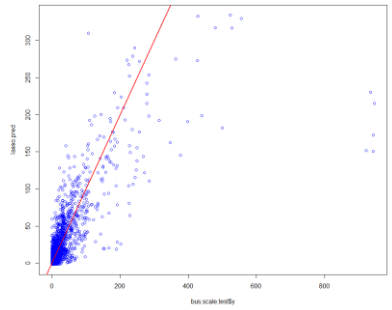
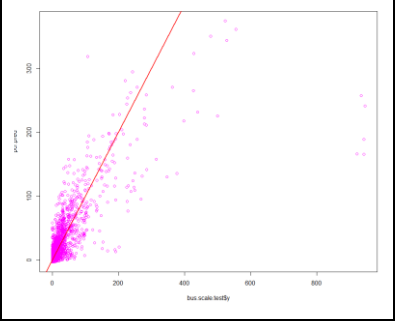
하지만, 각 주성분에 설명변수가 주는 영향이 뚜렷한 것이 아니고, 비슷한 영향 중에서 조금 수치가 큰 것들을 이용해 의미를 부여했습니다. 따라서 주성분이 가지는 정확한 의미는 아닙니다.

이를 뒷받침하는 것은 다중선형회귀로 설명변수를 주성분으로 하여 적합했을 때, 갖는 계수들이 앞서 모든 설명변수를 이용했던 다중선형회귀와 Ridge, Lasso 의 각 설명변수의 계수들과 의미를 부여한 주성분계수들 간에 부호, 수치 면에서 큰 차이가 나는 것을 확인 할 수 있습니다.

```
> cor(pcr.fit$scores) #주성분들의 상관계수가 모두 0임을 확인, 다중공선성 해결
              Comp 1        Comp 2        Comp 3        Comp 4        Comp 5        Comp 6        Comp 7
Comp 1  1.000000e+00 -4.606438e-15 -2.607422e-14 -1.859629e-14  1.289127e-14 -1.279784e-14 -1.570448e-15
Comp 2 -4.606438e-15  1.000000e+00  1.186308e-14  2.179308e-14  1.066397e-14 -2.836937e-14  1.821703e-15
Comp 3 -2.607422e-14  1.186308e-14  1.000000e+00 -6.281335e-15  2.431752e-15 -1.476068e-15 -3.815728e-15
Comp 4 -1.859629e-14  2.179308e-14 -6.281335e-15  1.000000e+00  1.290115e-14 -6.244683e-15 -3.242772e-15
Comp 5  1.289127e-14  1.066397e-14  2.431752e-15  1.290115e-14  1.000000e+00 -7.256871e-15 -3.585567e-15
Comp 6 -1.279784e-14 -2.836937e-14 -1.476068e-15 -6.244683e-15 -7.256871e-15  1.000000e+00  2.324574e-15
Comp 7 -1.570448e-15  1.821703e-15 -3.815728e-15 -3.242772e-15 -3.585567e-15  2.324574e-15  1.000000e+00
> pcr.fit$loadings #주성분들의 계수

Loadings:
             Comp 1 Comp 2 Comp 3 Comp 4 Comp 5 Comp 6 Comp 7
bus_in                                                -1.000
dawn_ride     -0.414  0.394  0.542  0.591 -0.143
dawn_takeoff  -0.203 -0.293  0.152        -0.389  0.835
work_ride     -0.551  0.366  0.127 -0.647  0.343
work_takeoff  -0.411 -0.581  0.160 -0.228 -0.380 -0.521
day_ride      -0.457  0.207 -0.796  0.206 -0.269
day_takeoff   -0.325 -0.495         0.369  0.703  0.108

               Comp 1 Comp 2 Comp 3 Comp 4 Comp 5 Comp 6 Comp 7
SS loadings     1.000  1.000  1.000  1.000  1.000  1.000  1.000
Proportion Var  0.143  0.143  0.143  0.143  0.143  0.143  0.143
Cumulative Var  0.143  0.286  0.429  0.571  0.714  0.857  1.000
```



(4) 부분선택법 결론

Ridge	Lasso	Pcr
RMSE=21.43479	RMSE=21.43479	RMSE= 21.05421
		

Ridge, Lasso, PCR 세 가지 기법을 사용하였고, 이 중 가장 RMSE가 작은 것은 PCR 입니다. 또한, Ridge와 Lasso는 7개의 설명변수를 모두 사용하였지만 PCR은 4개의 주성분을 사용하였습니다. Ridge와 Lasso가 주로 사용되는 가정이 제주도 퇴근시간 승차인원 데이터에는 잘 맞지 않았고, 몇 개의 주성분을 통해 데이터가 잘 설명된다는 PCR의 가정이 잘 맞음을 확인 할 수 있었습니다.

따라서, RMSE는 낮지만 모든 변수를 다 사용하며 다중공선성이 있는 다중선형회귀모델보다 PCR을 이용한 모델이 더 적합합니다. 따라서 선형모델에서는 PCR이 가장 적합한 모델입니다.

1.4. 비선형모델

데이터 탐색에서 하차 관련 설명변수들과 퇴근시간 승차인원(반응변수) 사이의 비선형적인 관계를 포착할 수 있었습니다. 따라서 이 부분을 적합할 수 있으면서 승차 관련 설명변수들의 선형성 또한 포함할 수 있는 모델인 일반화가법모델(GAMs)를 사용하기로 결정했습니다.

(1) 다항식회귀와 평활스플라인

(i) 선정 이유

GAMs에서 하차관련 설명변수의 경우 비선형성을 반영하기 위해 다항식회귀와 평활스플라인을 사용해 적합을 할 것입니다. 다항식회귀의 경우 설명변수 전체 범위에 걸쳐 고차원 다항식을 적합하는 것입니다. 조각별 상수회귀 기법의 확장인 기저함수를 바탕으로 하는 평활스플라인은 다항식회귀보다 좀 더 유연하면서 데이터에 더 평활한 모델을 만들 수 있습니다. 따라서, 이 두 가지를 활용한 GAMs를 비교해 보고, 어떤 적합이 더 나은지 비교해 보도록 하겠습니다.

(ii) 조율 파라미터 결정

GAMs에 다항식회귀와 평활스플라인을 넣기 전에 각 설명변수마다 각 방식에서 어떤 조율 파라미터를 가지면 좋을지를 알기 위해 먼저 tuning을 해보도록 하겠습니다.

3-fold 교차검증 방식을 사용하여 다항식회귀에서 각 설명변수에 대한 일반화된 차수를 구하였습니다. 평활스플라인에서는 leave-one-out 방식을 사용하여 일반화된 조율 파라미터 λ를 구하였습니다.

	poly	SmoothingSpline
Dawn_takeoff	2차	6.879156
Work_takeoff	4차	10.99203
Day_takeoff	2차	13.18741

(2) GAMs

(적합하기에 앞서 시내와 시외버스 구분을 나타내는 bus_in 변수는 GAMs를 적합할 때 뺐습니다. 그 이유는 앞에서 분석을 하면서 bus_in 변수가 반응변수 y를 분별하는데 큰 영향력이 없다고 판단 했기 때문입니다.)

(i) 적합 및 예측

아래 결과는 세 개의 승차관련 변수를 선형으로, 세 개의 하차관련 변수를 다항식 또는 평활스플라인으로 GAMs에 적합한 것입니다.

구체적으로 보면 비선형 처리를 해준 설명변수에 대한 p-값들은 선형 상관관계의 귀무가설과 비선형 상관관계에서의 대립가설에 대응합니다. 그리고 선형 처리를 해준 설명변수에 대한 p-값들은 그 반대의 가설이 성립합니다. 6개의 설명변수 모두 유의함을 볼 수 있고, 이는 각 설명변수에 대해서는 비선형 또는 선형 항이 필요하다는 명백한 증거를 보여줍니다.

또한, 각각의 GAMs 적합에 대해서 test set에 대해 예측한 결과 다항식회귀를 이용한 RMSE=20.8781, 평활스플라인을 이용한 RMSE=20.87542로 평활스플라인 쪽이 아주 조금 더 좋은 것을 알 수 있습니다. 하지만, 작은 차이이기 때문에 성능 면에서 큰 차이가 없음을 알 수 있습니다.

```
> summary(gam.fit.poly1)

Call: gam(formula = y ~ poly(dawn_takeoff^2) + poly(work_takeoff^4) +
    poly(day_takeoff^2) + dawn_ride + work_ride + day_ride, data = bus.ex.tra
Deviance Residuals:
      Min        1Q    Median        3Q       Max
-150.9453   -1.1663    0.4620    0.8553  838.7878

(Dispersion Parameter for gaussian family taken to be 219.2736)

    Null Deviance: 17073777 on 23430 degrees of freedom
Residual Deviance: 5136265 on 23424 degrees of freedom
AIC: 192803.9

Number of Local Scoring Iterations: 2

Anova for Parametric Effects
                     Df  Sum Sq Mean Sq F value    Pr(>F)
poly(dawn_takeoff^2)    1 2435915 2435915 11109.0 < 2.2e-16 ***
poly(work_takeoff^4)    1 1419756 1419756  6474.8 < 2.2e-16 ***
poly(day_takeoff^2)     1  808067  808067  3685.2 < 2.2e-16 ***
dawn_ride               1 4703508 4703508 21450.4 < 2.2e-16 ***
work_ride               1 1330857 1330857  6069.4 < 2.2e-16 ***
day_ride                1 1239409 1239409  5652.3 < 2.2e-16 ***
Residuals           23424 5136265     219
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> summary(gam.fit.spline)

Call: gam(formula = y ~ s(dawn_takeoff, 6.879156) + s(work_takeoff,
    10.99203) + s(day_takeoff, 13.18741) + dawn_ride + day_ride +
    work_ride, data = bus.ex.train)
Deviance Residuals:
      Min        1Q    Median        3Q       Max
-146.5733   -1.2944    0.2991    0.8742  827.3720

(Dispersion Parameter for gaussian family taken to be 206.6765)

    Null Deviance: 17073777 on 23430 degrees of freedom
Residual Deviance: 4835392 on 23395.94 degrees of freedom
AIC: 191445.6

Number of Local Scoring Iterations: 5

Anova for Parametric Effects
                                 Df  Sum Sq Mean Sq  F value    Pr(>F)
s(dawn_takeoff, 6.879156)         1 6277978 6277978 30375.86 < 2.2e-16 ***
s(work_takeoff, 10.99203)         1  344307  344307  1665.92 < 2.2e-16 ***
s(day_takeoff, 13.18741)          1   77550   77550   375.22 < 2.2e-16 ***
dawn_ride                         1 2828958 2828958 13687.85 < 2.2e-16 ***
day_ride                          1 2163265 2163265 10466.91 < 2.2e-16 ***
work_ride                         1   47556   47556   230.10 < 2.2e-16 ***
Residuals                     23396 4835392     207
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Anova for Nonparametric Effects
                             Npar Df Npar F     Pr(F)
(Intercept)
s(dawn_takeoff, 6.879156)        5.9 98.907 < 2.2e-16 ***
s(work_takeoff, 10.99203)       10.0 82.364 < 2.2e-16 ***
s(day_takeoff, 13.18741)        12.2 63.673 < 2.2e-16 ***
dawn_ride
day_ride
work_ride
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

> bus.ex.test<-as.data.frame(bus.ex.test) > poly.pred<-predict(gam.fit.poly1, newdata=bus.ex.test) > rmse(bus.ex.test$y, poly.pred) [1] 20.8781	> spline.pred<-predict(gam.fit.spline, newdata=bus.ex.test) > rmse(bus.ex.test$y, spline.pred) [1] 20.87542

(ii) GAMs 그래프로 보는 데이터 해석

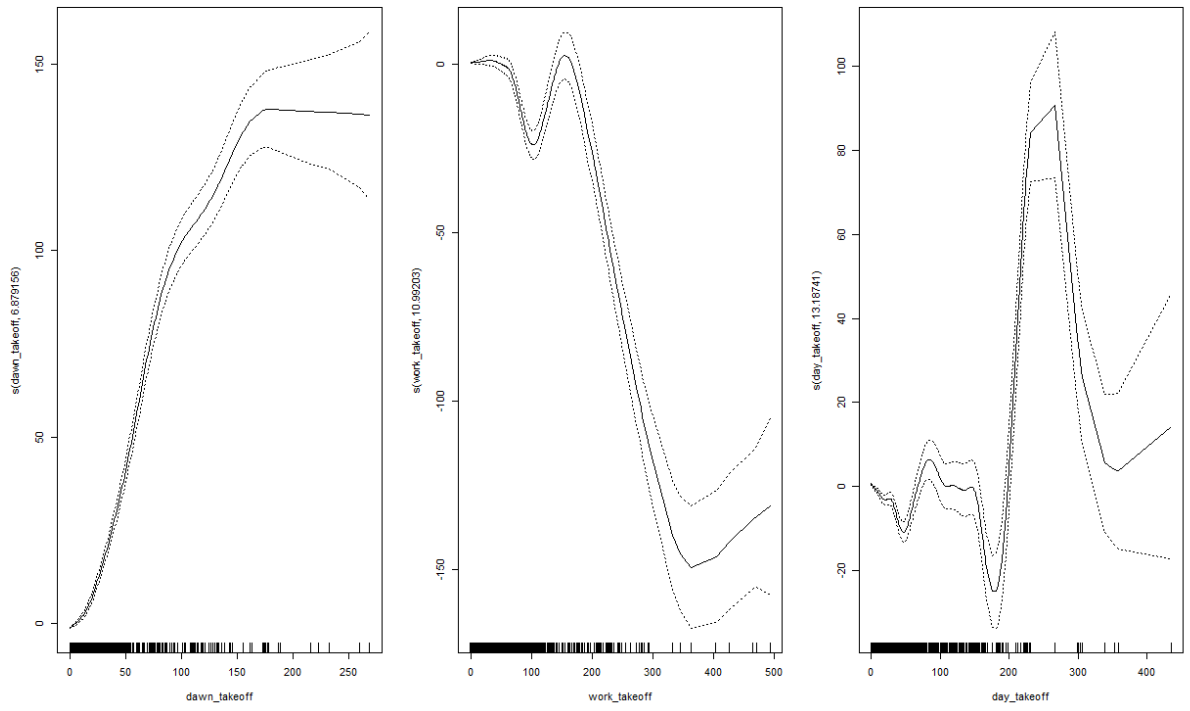
평활스플라인으로 적합한 모델에 대해서 그래프를 살펴보도록 하겠습니다.

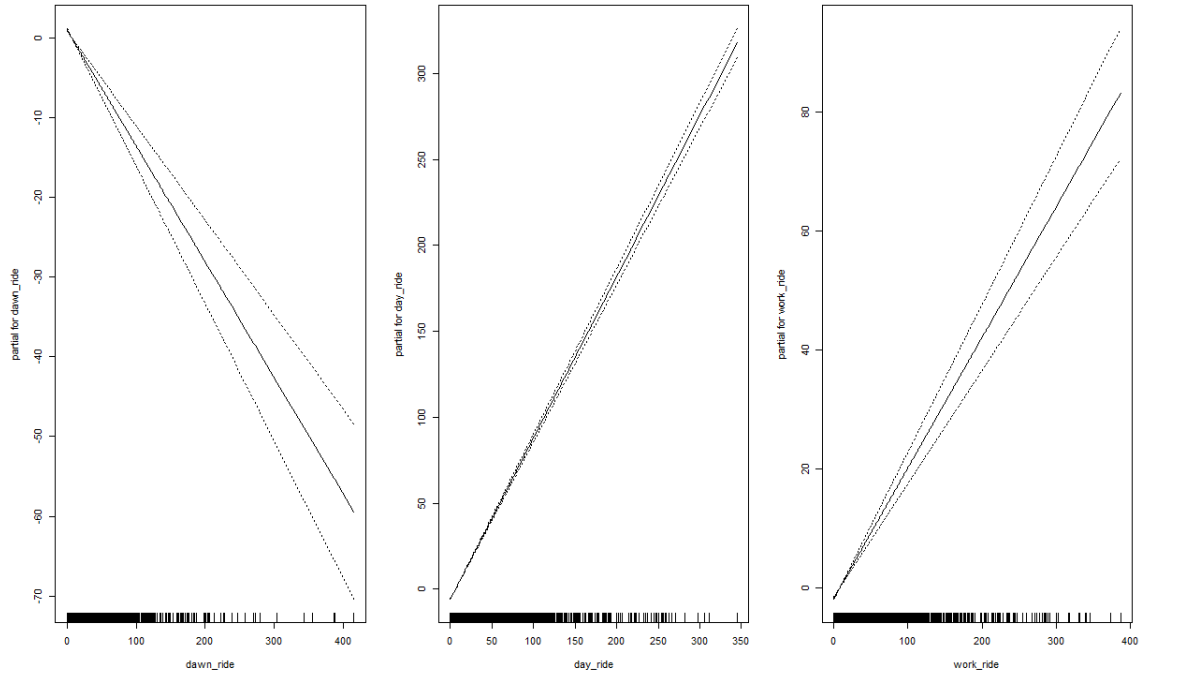
다음 그래프는 각 설명변수와 반응변수 사이의 관계를 보여주는 그래프입니다 적합된 함수와 조각별 표준오차를 나타냅니다 위의 세 개의 그래프는 하차와 관련된 변수들인 dawn_takeoff, work_takeoff, day_takeoff의 평활 스플라인입니다. 아래 세 개의 그래프는 승차와 관련된 변수들인 dawn_ride, work_ride, day_ride의 단순성형회귀직선 입니다.

Dawn_takeoff는 점차 올라가는 방향의 곡선이고, work_takeoff는 점차 내려가는 방향의 곡선입니다. Day_takeoff가 가장 굴곡이 심한데 200~300 사이에 급격히 높아지는 추세를 가지고 있는 것을 볼 수 있습니다.

Dawn_ride는 반응변수와 음의 관계를 가지고 있으며 work_ride와 Dawn_takeoff는 양의 관계를 가지고 있음을 알 수 있습니다.

<GLMs with Smoothing Spline 결과>





1.5. Tree 기반 모델

데이터 탐색에서 데이터가 선형으로 보이기도 하고, 비선형으로 보이기도 하였습니다. 따라서 선형도, 비선형도 아닌 일반적인 데이터 분포로서 볼 수 있으며 이러한 데이터에 알맞은 기법으로 Tree 기반 모델을 사용하였습니다.

(1) Regression Tree

(i) 선정 이유

회귀트리 1개는 데이터에 대해 설명하기 쉽지만 예측 정확도가 떨어집니다. 따라서 Regression tree를 통해 예측력보다는 퇴근시간 버스 승차인원에 대한 설명변수들의 해석을 위해 사용하였습니다.

(ii) 적합 및 예측

적합한 Regression tree를 train set에 대해 예측한 결과, RMSE=19.85481 이 나옴을 알 수 있습니다. 이는 앞서 다중선형회귀모델, 변수선택모델(Rdige, Lasso, PCR)의 RMSE에 비해 낮음을 알 수 있습니다.

Tree 모델의 예측력이 떨어짐에도 불구하고 1개의 tree에 대해 RMSE가 낮게 나온 것은 데이터가 선형, 비선형적인 모양을 가진 것이 아닌 일반적인 분포를 따르기 때문일 것이라고 추측할 수 있습니다.

(iii) Regression Tree를 통한 데이터 해석

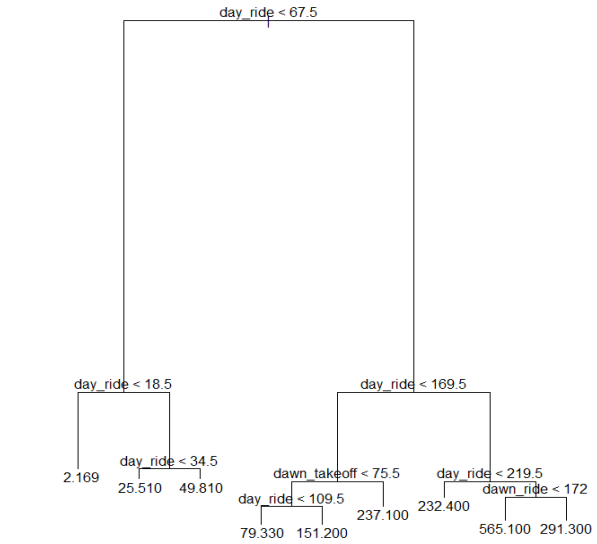
Regression tree를 train set의 모든 설명변수에 대해 적합한 결과 day_ride, dawn_takeoff, dawn_ride 3가지의 변수들만 사용되었고, 나열된 순서대로 데이터에 중요한 변수임을 알 수 있습니다. Day_ride(8시~10시 승차인원)이 많을수록 퇴근시간 승차인원이 많아지는 것을 볼 수 있습니다. 따라서 Day_ride와 퇴근시간 승차인원이 선형성을 갖는 것을 알 수 있습니다.

또한, 낮 승차인원이 많은 범주 day_ride>=169.5 일 때를 주목해보면, dawn_ride(6~8시 탑승인원)이 오히려 적을 때 퇴근시간 승차인원이 더 큰 것을 알 수 있습니다. 따라서 퇴근시간 승차인원이 높은 정류소는 새벽에 유동인구가 적지만 낮에는 급격히 많아지는 곳에 위치해 있을 것입니다.

<결과>

```
> tree.rfit<-tree(y~., bus.ex.train)
> summary(tree.rfit)

Regression tree:
tree(formula = y ~ ., data = bus.ex.train)
Variables actually used in tree construction:
[1] "day_ride"      "dawn_takeoff" "dawn_ride"
Number of terminal nodes:  9
Residual mean deviance:  193.5 = 4533000 / 23420
Distribution of residuals:
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
-382.1000   -2.1690   -2.1690    0.0000   -0.1694  759.8000
```

```
> tree.rfit
node), split, n, deviance, yval
      * denotes terminal node

 1) root 23431 17070000   6.581
   2) day_ride < 67.5 23027  3539000   4.224
     4) day_ride < 18.5 21644   803800   2.169 *
     5) day_ride > 18.5 1383  1214000  36.370
      10) day_ride < 34.5 765   394400  25.510 *
      11) day_ride > 34.5 618   617800  49.810 *
   3) day_ride > 67.5 404  6112000 141.000
     6) day_ride < 169.5 347  2486000 114.100
      12) dawn_takeoff < 75.5 317  1593000 102.400
        24) day_ride < 109.5 215   399600  79.330 *
        25) day_ride > 109.5 102   836600 151.200 *
      13) dawn_takeoff > 75.5 30   395600 237.100 *
     7) day_ride > 169.5 57  1851000 304.500
      14) day_ride < 219.5 29   402900 232.400 *
      15) day_ride > 219.5 28  1141000 379.300
        30) dawn_ride < 172 9   569200 565.100 *
        31) dawn_ride > 172 19   113400 291.300 *
```

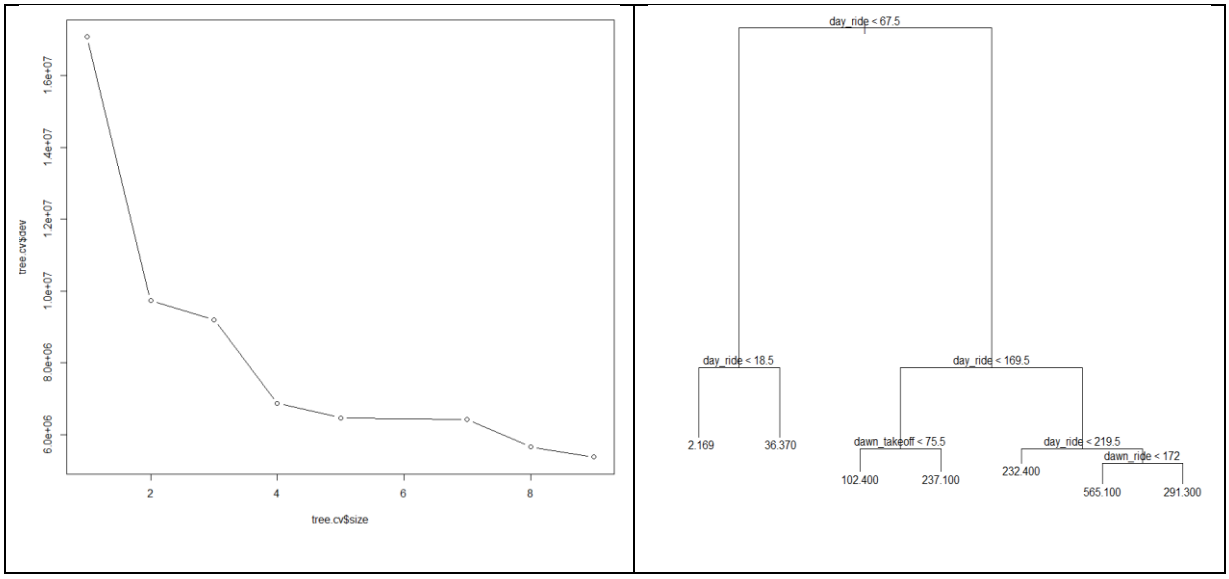



```
> tree.pred.r<-predict(tree.rfit,newdata= bus.ex.test)
> rmse(bus.ex.test$y, tree.pred.r)
[1] 19.85481
```

(iv) Pruning(가지치기)

Regression Tree 모델의 과적합 방지를 위해 가지치기를 사용할 수도 있습니다. 교차검증을 사용해 Tree의 size를 6으로 지정해 주었고, 그에 따른 tree 그래프를 그렸습니다. 가지치기한 모형을 test set에 대해 예측한 결과, rmse=20.74136으로 가지치기 전보다 정확성이 떨어집니다.

하지만, 가지치기 전의 tree의 rmse가 매우 낮아서 과적합이 될 것이라 우려되지 않기 때문에 가지를 친 모형을 이용하지는 않을 것입니다.



(2) RandomForeest

(i) 선정 이유

Regression Tree는 높은 분산을 가진다는 문제점이 있습니다. Bagging은 붓스트랩 방식을 이용해서 여러 개의 훈련셋을 만들어 각각에 대해 tree를 적합하고, 그 tree들을 평균을 내어 결합하는 방식입니다. 이렇게 함으로서 상당한 정확도 향상을 이룰 수 있음이 입증되었습니다.

하지만 bagging은 트리들간의 상관성이 높습니다. 따라서 변수를 모두 사용하는 것이 아닌 일부만을 부분추출하여 사용하는 Random Forest를 이용할 것입니다.

이를 통해 정확도를 높이고, 결합으로 인한 해석은 어렵지만 변수의 중요도를 살펴보도록 하겠습니다.

(ii) 적합 및 예측

Mtry, 즉 추출할 변수의 개수를 정하기 위해 tuneRF 함수를 이용했습니다. Oob error가 mtry=3에서 가장 작으므로 mtry를 3으로 지정해주어 적합을 하도록 하겠습니다.

Test set에 대해 예측한 결과, RMSE=18.3175 입니다. Regression Tree에 비해 RMSE가 낮게 나온 것을 확인할 수 있습니다.

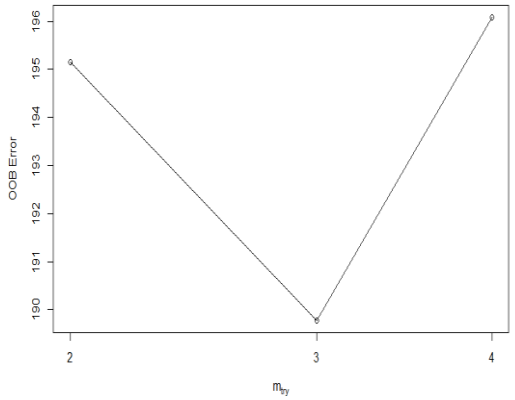
(iii) 변수 중요도

변수 중요도에 대한 결과에서 %IncMSE는 해당 변수를 Pumutation(순서섞기)한 이후 MSE 증가분을 의미하고, IncNodePurity는 해당 변수를 사용한 노드의 MSE 감소의 합의 평균을 의미합니다.

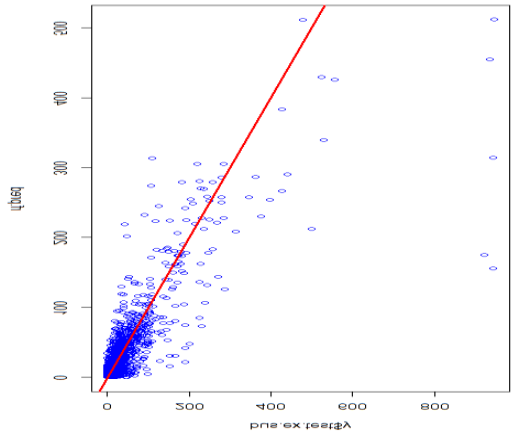
%IncMSE를 기준으로 보면 day_takeoff, work_takeoff, dawn_ride, day_ride 순으로 변수가 중요함을 알 수 있습니다. IncNodePurity 기준으로는 day_ride, work_ride, dawn_ride, dawn_takeoff 순입니

다.

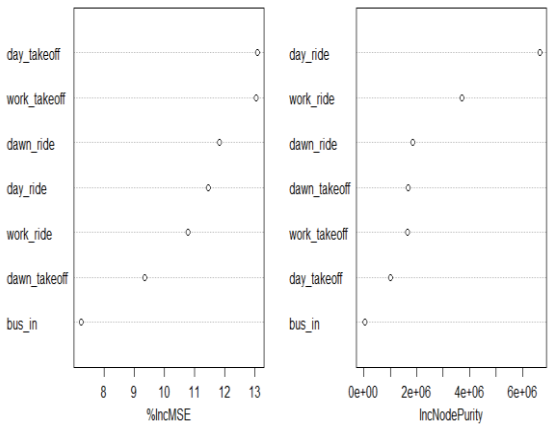
```
> set.seed(1)
>
> features<-setdiff(names(bus.ex.train), "y")
>
> tuneRF(x=bus.ex.train[features], y=bus.ex.train$y,
+
+        ntreeTry=300, stepFactor=1.5, improve=0.01)
mtry = 2  OOB error = 195.1586
Searching left ...
Searching right ...
mtry = 3        OOB error = 189.7832
0.02754375 0.01
mtry = 4        OOB error = 196.0789
-0.03317321 0.01
  mtry OOBError
2    2 195.1586
3    3 189.7832
4    4 196.0789
```




```
> rf.fit=randomForest(y~.,data=bus.ex.train,mtry=3,ntree=300,importance=TRUE)
> rf.fit

Call:
 randomForest(formula = y ~ ., data = bus.ex.train, mtry = 3,      ntree = 300, importance = TRUE)
               Type of random forest: regression
                     Number of trees: 300
No. of variables tried at each split: 3

          Mean of squared residuals: 189.9496
                    % Var explained: 73.93
>
> rf.pred<-predict(rf.fit, bus.ex.test)
> rmse(rf.pred, bus.ex.test$y)
[1] 18.3175
> #plot
> plot(rf.pred, bus.ex.test$y, col=c(4))
> abline(0,1, col="red", lwd=2)
```




```
> importance(rf.fit)
                %IncMSE IncNodePurity
bus_in         7.247598      37132.82
dawn_ride     11.828369    1841092.33
dawn_takeoff   9.344276    1688790.26
work_ride     10.792092    3707230.19
work_takeoff  13.043799    1642384.51
day_ride      11.464667    6632693.68
day_takeoff   13.083060    1018874.60
>
> varImpPlot(rf.fit) #plot
```



1.6. 최종 결과

(1) 적합모델 선정

선형모델(다중선형회귀, Ridge, Lasso, PCR), 비선형모델(GAMs-Poly, GAMs-SmoothingSpline), Tree 모델(RegressionTree, RandomForest) 총 7가지 모델에 대해서 교차검증을 이용해 hyper parameter tuning을 하였고, train set에 적합했습니다. 적합 모형으로 validation set의 반응변수를 예측해 검증오차 RMSE를 구했습니다.

선형모델	다중선형회귀	Ridge	Lasso	PCR
RMSE	20.84169	21.43479	21.43479	21.05421
비선형+Tree	GAMs-poly	GAMs-spline	Reg-Tree	RandomForest
RMSE	20.8781	20.87542	19.85481	18.3175

선형 모델 중 PCR이 가장 적합하다고 판단했습니다. 그 이유는 RMSE가 가장 낮지는 않지만 4개의 주성분을 이용해 다중선형회귀와 비슷한 설명력과 Ridge, Lasso에 비해 더 높은 예측력을 가지고 있다는 것을 알 수 있었습니다.

비선형 모델은 다항식회귀와 평활스플라인 방식 중 큰 차이는 없었으나 평활스플라인의 유연성과 과적합 방지가 더 일반적으로 좋은 성능을 낼 것이라 판단해 평활스플라인 방식을 선택했습니다. 평활 스플라인 방식과 단항식회귀 방식을 가법적으로 사용해 비선형 모델보다 더 높은 예측력을 가질 수 있었습니다.

Tree 기반 모델은 데이터의 분포가 선형도 비선형도 아닌 일반적인 분포라는 생각에 기반하였습니다. Regression Tree 1개가 다른 선형 모델과 비선형 모델에 비해 예측력이 높은 것을 통해 가정이 맞다는 것을 증명했습니다. 이를 바탕으로 Random Forest로 모델을 확장했고 가장 예측력을 높일 수 있었습니다.

Random Forest는 임의로 나누어준 train, test set에 대해서만 성능이 좋을 수 있습니다. 따라서 교차검증을 통해 Random Forest의 보편적인 성능을 알아보려 합니다. 또한, 과적합이 일어나는지도 확인해 볼 것입니다. 그 결과 5 fold 교차검증을 통해 나눈 각각의 5개의 rmse의 평균이 10.9887이 나왔습니다. 임의로 나누어준 data set 이외에도 성능이 좋은 것을 확인 할 수 있습니다..

```
> apply(mean.rmse, 2, mean)
mean.rmse
 10.98987
```

결론적으로 제주도 버스 퇴근시간 승차인원 데이터는 Random Forest 모델에 의해 가장 잘 적합됩니다.

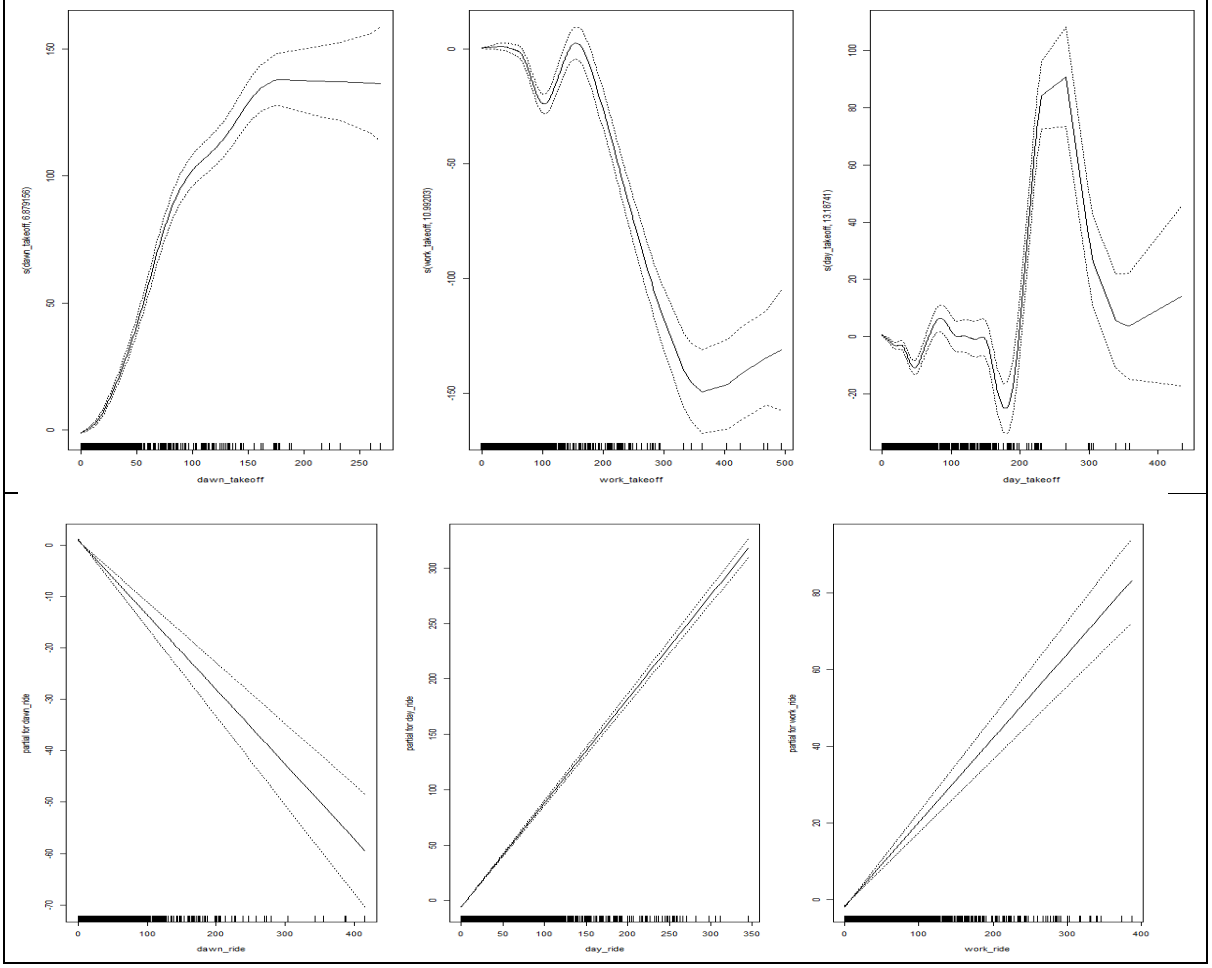
(2) 제주도 퇴근시간 승차 인원 데이터 해석

(i) 결과 요약

선형 모델에서 각 모델마다 각 설명변수의 계수 추정치를 아래 표에 요약했습니다. 이와 더불어 비선형 모델에서 적합한 GAMs의 각 설명변수와 반응변수와의 관계를 그린 적합선을 같이 보도록 하겠습니다.

설명변수	Dawn_ride	Work_ride	Day_ride	Dawn_takeoff	Work_takeoff	Day_takeoff
다중선형 회귀	-0.19519	0.26661	0.94503	0.80793	-0.21981	
Ridge	-0.007805	0.24904	0.71044	0.50825	-0.03258	0.0375
Lasso	-0.1787	0.247501	0.950805	0.79971	-0.18152	-0.04075

<GLMs with Smoothing Spline 결과>



아래 표는 다중선형회귀, PCR Tree 기반 모델에서 각 변수의 중요도를 요약한 표입니다.

	1	2	3	4	5
다중선형회귀	Day_ride	Dawn_takeoff	Work_takeoff	Work_ride	Dawn_ride
PCR(순서상관 없음)	Dawn_ride	Dawn_takeoff	Work_ride	Work_takeoff	Day_rid
%IncMSE	Day_takeoff	Work_takeoff	Dawn_ride	Day_ride	
IncNodePurity	Day_ride	Work_ride	Dawn_ride	Dawn_takeoff	

(ii) 결과 분석

Work_ride, day_ride, dawn_takeoff는 계수추정치가 양수이며 모두 유의하고 계수추정치의 분산 또한 크지 않았습니다. 적합선 또한 양의 관계를 가짐을 알 수 있습니다. 좀 더 구체적으로 말해보면, 낮에 타는 사람이 많을수록, 새벽에 하차하는 사람이 많을수록, 출근시간에 탑승하는 사람이 많을수록 퇴근시간 승차인원은 늘어납니다.

Dawn_ride, work_takeoff는 계수추정치가 음수이며 적합선 또한 음의 관계를 가짐을 알 수 있습니다. 좀 더 구체적으로 말해보면, 새벽에 탑승하는 사람이 많을수록, 출근시간에 하차하는 사람이 많을수록 퇴근시간 승차인원은 감소합니다.

구체적으로 해석하기 위해 잠시 Ⅰ에서 데이터를 그래프로 그리며 탐색했던 내용을 상기합니다.

새벽, 출근시간, 낮 시간에서의 모든 승차와 하차 변수들과 퇴근시간 승차인원은 대체로 양의 상관성을 띄는 것을 보았습니다. 그리고 그 이유는 버스 정류소가 있는 장소에서 시간에 상관없이 그 만큼의 유동인구가 버스를 이용하기 때문이었습니다. 이 부분이 선형 모델에서 잘 드러나 있습니다. (이 때, 유동인구는 회사원뿐만이 아니라 학생, 주부 등 여러 계층의 사람들을 의미합니다.) 그리고 이 상관성을 A 패턴이라 부르겠습니다.

하지만, 선형모델에 있어서는 이상치일 수 있으며 퇴근시간 승차인원이 매우 높은 데이터들이 있었습니다. 이에 선형모델에서 모델을 적합할 때 이 부분들에 큰 영향을 받아 양의 상관성을 띄는 전체적인 패턴에서 벗어나 또 다른 패턴을 만들어 내었습니다. 이 패턴을 B라 부르겠습니다.

출근시간 승차인원과 낮에 승차인원은 A 패턴이 잘 드러나는 변수들입니다. 즉, 버스 정류소 가있는 곳의 유동인구의 수만큼 버스를 낮시간이든 퇴근시간이든 이용을 한다는 것입니다. 특

히, 낮에 승차인원은 퇴근시간 승차인원이 매우 높은 데이터들이 그래프에서 가장 오른쪽에 있기 때문에 선형성이 강한 변수입니다. 따라서, 가장 특징이 두드러지는 변수였기 때문에 퇴근시간 승차인원을 예측하는 데 가장 큰 영향력을 주었다고 생각합니다. 변수 중요도 표에서도 day_ride가 가장 중요한 변수임은 쉽게 파악할 수 있습니다. 출근시간 승차인원은 낮에 승차인원에 비해 승차인원이 매우 높은 데이터가 그래프상 오른쪽에 애매하게 있어 그 중요도가 낮아졌다고 생각합니다.

출근시간 하차인원은 B 패턴이 드러나는 변수입니다. 즉, 출근 시간에 버스를 이용해 목적지에 하차하면 퇴근시간에는 반대 정류소를 이용할 것입니다. 출근시간 하차인원은 그래프를 보면 퇴근시간 승차인원이 그래프상 왼쪽에 치우쳐져 있어 이상치의 영향을 많이 받은 것을 알 수 있습니다. 비선형 모델에서 평활스플라인의 적합선 또한 비슷한 형상을 띄고 있습니다. 하지만, A 패턴 또한 없어진 게 아닌 남아있으면서 이상치들의 영향을 받았기 때문에 뚜렷한 특징을 가지고 있지 않는 것으로 판단해 중요도가 비교적 낮아졌다고 생각합니다.

추가적으로, 새벽시간 승차인원은 음의 상관성을 띄는데 이상치의 영향을 받아서이기도 하며 유동인구 때문이기도 합니다. 이상치들이 주로 있을 것이라 추론한 도심, 회사가 많이 있는 곳은 출근시간, 낮이 되어야 유동인구가 많아지는 곳입니다. 따라서, 새벽에는 오히려 다른 정류소들보다 유동인구가 적다는 것입니다. 이를 뒷받침하는 근거로 Regression Tree 모델에서 낮에 승차인원이 매우 많은 day_ride>=169.5이면서 dawn_ride(6~8시 탑승인원)가 오히려 적을 때 퇴근시간 승차인원이 더 컸습니다.

다만, 해석하기 어려운 변수도 존재합니다. Dawn_takeoff의 경우 A 패턴과 B 패턴이 모두 섞이어 정확한 근거를 제시하기 어렵습니다. 선형모델에서는 중요한 계수이며 양의 상관관계를 띄는 이유는 A 패턴이 B 패턴보다 더 강하기 때문일 것입니다. 하지만, Tree 기반 모델에서는 dawn_takeoff의 중요성이 현저하게 떨어지는 것으로 보아 일반적인 분포를 잘 다루지 못하는 선형성의 한계로 보입니다.

day_takeoff의 경우 계수추정치가 양수, 음수 모두 있는 것을 알 수 있습니다. 그리고 적합선 또한 일정한 관계를 가지지 않음을 알 수 있습니다. 확연한 특징을 가지고 있지 않아 해석이 어렵습니다.

2.분류분석

2.1. 데이터 탐색

(i) 분석 목적

퇴근 시간 버스 승차인원이 많은 정류소와 퇴근 시간 버스 승차인원이 적거나 보통인 정류소를 나누어 퇴근 시간 승차인원이 많은 정류소들의 특징에 대해 알아볼 것입니다.

(ii) 모델 기법 선택

로지스틱회귀모형으로 예측확률을 이용하여 분류해보고고, 선형·이차판별분석, 서포트벡터분류기를 통해 선 또는 이차선 경계를 이용해 분류해 볼 것입니다. Tree기반 구조를 통해서도 분류해 볼 것입니다.

(iii) 데이터 처리

퇴근시간 승차인원을 질적변수로 지정해 줍니다. 이 때 "퇴근 시간 승차인원이 많다"의 기준은 60으로 정했습니다. 즉, 18시~20시 사이에 한 정류소에서 60명의 인원 이상이 탑승할 경우입니다. 60명을 기준으로 둘 경우 정류장에 사람이 많아지고, 버스 혼잡함이 증가하며 교통체증도 늘어날 것이라 생각했기 때문입니다.

그 이유는 두 시간 동안 60명(1시간 동안 30명)의 사람들이 일정한 간격으로 버스 정류장에 오는 것이 아닌 비슷한 시간에 한꺼번에 올 경우가 많다고 생각했습니다. 또한, 33471개의 관측치 중 827개의 관측치만이 기준을 넘는 것을 통해 버스정류소의 위치가 회사가 많이 위치해 있는 한정된 곳이라는 점에서 부합하다고 생각했습니다.

(iv) Train, test set 분리

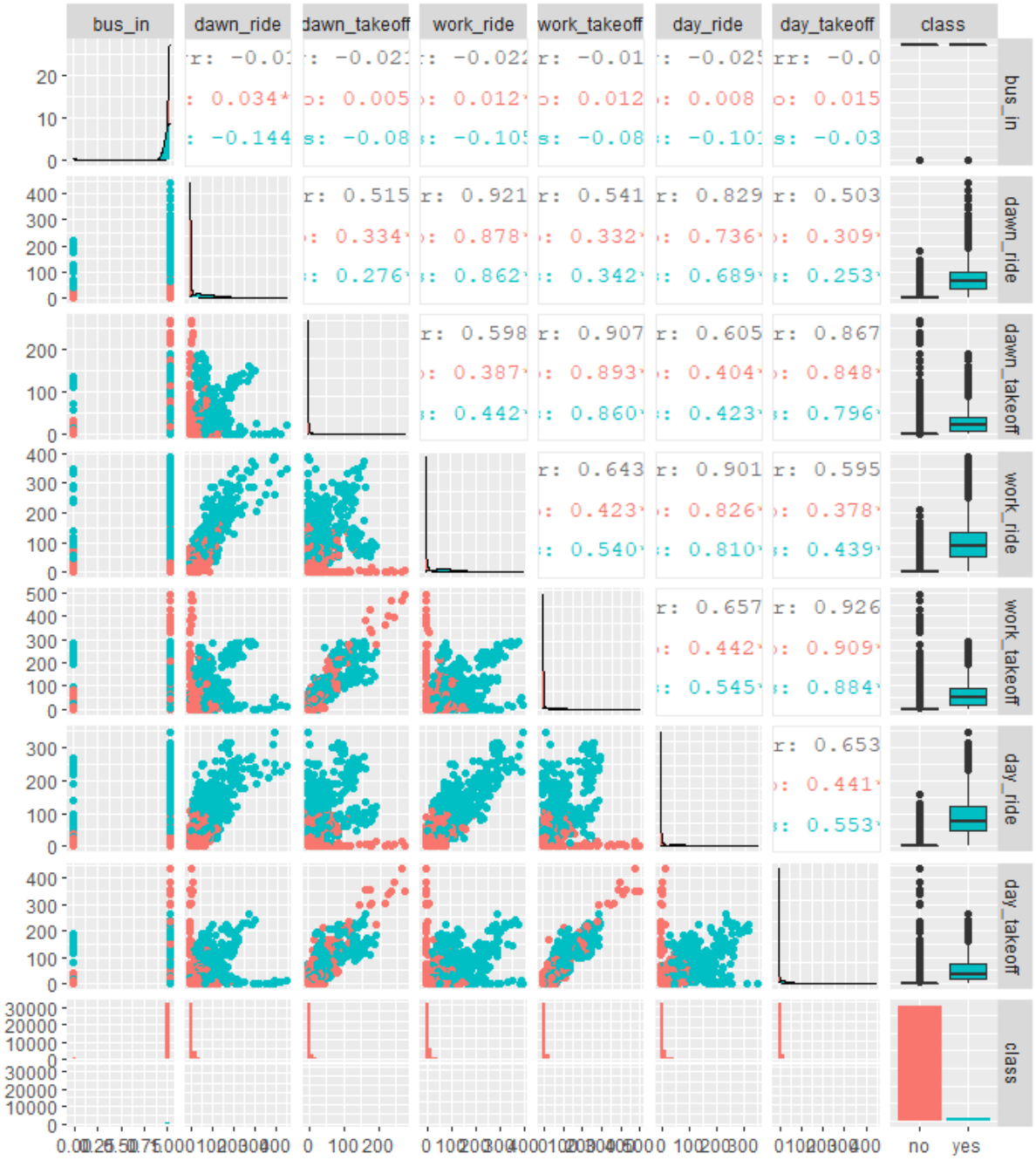
분류 분석에 앞서 데이터를 train, validation set으로 나누어 줍니다. createDataPartition 함수를 사용하여 "퇴근시간 승하차 인원이 많은 정류소"가 train, validation set에 각각 몰리는 것을 방지했습니다. 각 train, validation set에 "퇴근시간 승하차 인원이 많은 정류소"를 일정한 비율로 넣어 주었습니다.

```
> set.seed(1)
> idx<-createDataPartition(bus.ex.class$class, p=0.7, list=FALSE)
> bus.ex.ctrain<-bus.ex.class[idx,]
> bus.ex.ctest<-bus.ex.class[-(idx),]
> table(bus.ex.ctrain$class); table( bus.ex.ctest$class)

   no   yes
22851   579

  no  yes
9793  248
```

(v) Ⅰ-4-(3)과 같은 데이터 시각화

분류를 통해 "퇴근시간 승차인원이 많은 정류소"의 세세한 특징에 대해 더 알아봅니다.

Ⅰ-4-(3) 내용과 일부 중복적이지만 60을 기준으로 나눈 분류 데이터를 탐색하고 보이지 않았던 특징을 찾을 수 있을 거라는 점에 의의를 둡니다.



Box plot을 살펴보면 dawn_ride에 비해 work_ride, day_ride의 box가 높아진 것을 볼 수 있습니다. 시간이 지나면서 정류소 근처에 유동인구가 많아지는 특징을 가지고 있습니다. Work_ride와 day_ride 관계 그래프를 보면 "승차 인원이 많은 정류소"는 그렇지 않은 정류소에 비해 낮 시간에

승차하는 인원이 모두 많았습니다.

승차 관련 변수들에 비해 하차 관련 변수들은 대체적으로 box가 낮은 것을 볼 수 있습니다. 그 이유는 퇴근시간 승차인원이 새벽, 출근시간, 낮 시간에 반대 정류소에서 하차했기 때문입니다. 즉, 방향 때문에 같은 정류소에서 하차와 승차를 같이 하지 않으므로 상대적으로 승차 대비 하차 인원이 적은 것입니다.

설명변수 간의 관계를 나타낸 그래프를 보면 기본적으로 "승차 인원이 많은 정류소"는 출근시간, 낮 모두 "그렇지 않은 정류소"보다 인원이 많은 것을 볼 수 있습니다. 하지만, 새벽 시간의 경우 그렇지 않은 정류소와 완연히 분리되지 않고, 그렇지 않은 정류소보다 오히려 승차 또는 하차 인원이 적은 것을 볼 수 있습니다.

이 네 가지 특징을 통해 "승차 인원이 많은 정류소"는 도심, 또는 회사가 밀집해 있는 곳에 있을 것으로 추론할 수 있습니다. 따라서 교통혼잡은 이 곳에서 주로 발생할 것임을 알 수 있습니다.

2.2. 로지스틱 회귀모델

(i) 선정 이유

반응변수가 질적 변수이기 때문에 회귀모형을 사용하기에는 적절하지 않습니다. 따라서 $\log(\hat{\pi}(x)/(1-\hat{\pi}(x)) = \alpha + \beta_1 x_1 + \beta_2 x_2 + \cdots\cdots + \beta_p x_p$ 형태인 로지스틱 회귀모델을 이용해 "퇴근시간 승차인원이 많은 정류소"에 대한 성공확률("퇴근시간 승차인원이 많은 정류소"일 확률)을 예측해 봅니다.

(ii) 적합 및 예측

Train set 을 이용해 로지스틱 회귀를 적합했습니다. 이 적합모델을 사용해 test set 의 성공확률을 예측합니다. 이 때, 임계치는 ROC 곡선을 통해 구합니다. 적절한 임계치를 구하는 함수를 이용해 임계치가 0.01395854 일 때 가장 좋은 민감성(sensitivity)와 특이도(specificity)를 가지는 것을 알 수 있습니다. 임계치가 이렇게 낮게 나온 이유는 타겟관측치들이 전체 관측치들에 비해 매우 적기 때문입니다.

예측한 결과, Accuracy=0.9417, Sensitivity=0.9418, Specificity=0.9395 로 매우 높게 나오는 것을 볼 수 있습니다. ("퇴근시간 승차인원이 많은 정류소"는 1 로 정해집니다.)

Bus_in, dawn_ride, work_ride, day_ride가 가장 유의하게 나왔습니다. 그 다음으로 dawn_ride, dawn_takeoff가 유의하며 work_takeoff, day_takeoff는 유의하지 않았습니다. 각 계수 추정치의 분산이 work_takeoff와 day_takeoff를 제외하면 높지 않은 것을 볼 수 있습니다.

```
> summary(logistic.fit)

Call:
glm(formula = class ~ ., family = "binomial", data = bus.ex.ctrain)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.6870  -0.0802  -0.0696  -0.0666   3.3744

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  -4.802508   0.342737 -14.012  < 2e-16 ***
bus_in       -1.332415   0.344058  -3.873 0.000108 ***
dawn_ride     0.011688   0.004303   2.716 0.006599 **
dawn_takeoff  0.023843   0.007575   3.148 0.001645 **
work_ride     0.014805   0.004390   3.372 0.000745 ***
work_takeoff -0.005522   0.005357  -1.031 0.302604
day_ride      0.057979   0.003796  15.273  < 2e-16 ***
day_takeoff   0.004287   0.005130   0.836 0.403282
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5428.7  on 23429  degrees of freedom
Residual deviance: 1837.5  on 23422  degrees of freedom
AIC: 1853.5

Number of Fisher Scoring iterations: 8
```

```
> confusionMatrix(as.factor(logistic.pred.class),
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  9223   15
       yes  570  233

               Accuracy : 0.9417
                 95% CI : (0.937, 0.9462)
    No Information Rate : 0.9753
    P-Value [Acc > NIR] : 1

                  Kappa : 0.4216

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.9418
            Specificity : 0.9395
         Pos Pred Value : 0.9984
         Neg Pred Value : 0.2902
             Prevalence : 0.9753
         Detection Rate : 0.9185
   Detection Prevalence : 0.9200
      Balanced Accuracy : 0.9407

       'Positive' Class : no
```

2.3. 선형·이차판별분석

(1) 선형판별분석(Linear discriminant Analysis)

(i) 선정 이유

로지스틱회귀모델은 클래스들이 잘 분리될 때 로지스틱 회귀모델에 대한 모수 추정치는 불안정합니다. 선형·이차판별분석은 이러한 문제가 없기 때문에 선형판별분석을 이용하려 합니다.

다만, 선형판별분석은 "X(설명변수의 벡터형태)는 클래스 특정 평균벡터와 공통의 공분산행렬을 가지는 다변량가우스분포를 따른다. 또한, 다변량 가우스분포는 각 설명변수가 설명변수들의 각 쌍 간에 상관성이 있는 1차원 정규분포를 따른다." 라는 가정 하에 적합하는 모델입니다.

제주도 퇴근시간 승차인원의 설명변수들은 다변량가우스분포를 따르지 않음을 데이터 탐색을 통해 확인했습니다. 따라서 판별분석모델 적합을 시도만 할 뿐, 적합한 모델로서 미리 고려하지 않는다는 점을 염두에 두어야 합니다.

(ii) 적합 및 예측

Prior probabilities of group 을 통해 훈련 관측치들의 약 97.5%가 no 이며 약 2.5%가 yes 임을 알 수 있습니다. Group means 를 통해 한 관측치가 dawn_ride 가 74 명, dawn_takeoff 가 34 명, work_ride 가 102 명, work_takeoff 가 68 명, day_ride 가 90 명, day_takeoff 가 54 명에 가깝거나 그 이상이면 yes 로 분류될 것입니다. 따라서 -1.410142e-01x_1+ -2.018973e-05x_2+ 3.512373e-02x_3+1.252460e-02x_4+ -8.456620e-03x_5+ 5.457921e-02x_6+ 1.124714e-03x_7 분류기를 기준으로 yes 아니면 no 로 분류될 것입니다.

예측한 결과, Accuracy=0.9834, Sensitivity=0.9916, Specificity=0.6573 로 높게 나오는 것을 볼 수 있습니다. (임계치는 0.5 로 default 값을 이용했습니다.)

```
> lda.fit
Call:
lda(class ~ ., data = bus.ex.ctrain)

Prior probabilities of groups:
        no        yes
0.97528809 0.02471191

Group means:
       bus_in dawn_ride dawn_takeoff  work_ride work_takeoff day_ride day_takeoff
no  0.9833705  4.907619     1.778784   5.763555     3.693755  4.12017    3.073826
yes 0.9585492 74.827288    34.683938 102.132988    68.563040 90.73575   54.848014

Coefficients of linear discriminants:
                    LD1
bus_in       -1.410142e-01
dawn_ride    -2.018973e-05
dawn_takeoff  3.512373e-02
work_ride     1.252460e-02
work_takeoff -8.456620e-03
day_ride      5.457921e-02
day_takeoff   1.124714e-03
```

```
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  9711   85
       yes   82  163

               Accuracy : 0.9834
                 95% CI : (0.9807, 0.9858)
    No Information Rate : 0.9753
    P-Value [Acc > NIR] : 2.082e-08

                  Kappa : 0.6527

 Mcnemar's Test P-Value : 0.877

            Sensitivity : 0.9916
            Specificity : 0.6573
         Pos Pred Value : 0.9913
         Neg Pred Value : 0.6653
             Prevalence : 0.9753
         Detection Rate : 0.9671
   Detection Prevalence : 0.9756
      Balanced Accuracy : 0.8244

       'Positive' Class : no
```

(2) 이차판별분석(Quadratic Discriminant Analysis)

(i) 선정 이유

선형판별분석의 제한된 가정이 완화되어 이차판별분석에서는 "각 클래스의 관측치들이 가우스 분포를 따르고, 각 클래스가 자체 공분산행렬을 갖는다."라는 가정을 가지게 됩니다. 클래스들의 공분산행렬이 공통이 아닌 것으로 여겨지기 때문에 QDA를 사용하는 것이 더 나은 결과를 줄 수 있을 거라 생각했습니다. (훈련셋이 크므로 분류기의 분산이 우려사항이 아닙니다.)

(ii) 적합 및 예측

Group means 는 LDA 와 거의 비슷합니다. 한 관측치가 dawn_ride 가 74 명, dawn_takeoff 가 34 명, work_ride 가 102 명, work_takeoff 가 68 명, day_ride 가 90 명, day_takeoff 가 54 명에 가깝거나 그 이상이면 yes 로 분류될 것입니다.

예측을 한 결과, Accuracy=0.9399, Sensitivity=0.9419, Specificity=0.8629 로 높게 나오는 것을 볼 수 있습니다. Accuracy 와 Sensitivity 는 줄어들었으나 Specificity 가 그에 비해 높아진 것을 확인할 수 있습니다. (임계치는 0.5 로 default 값을 이용했습니다.)

```
Call:
qda(class ~ ., data = bus.ex.ctrain)

Prior probabilities of groups:
        no        yes
0.97528809 0.02471191

Group means:
       bus_in dawn_ride dawn_takeoff  work_ride work_takeoff day_ride
no  0.9833705  4.907619     1.778784   5.763555     3.693755  4.12017
yes 0.9585492 74.827288    34.683938 102.132988    68.563040 90.73575
    day_takeoff
no     3.073826
yes   54.848014
```

```
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  9224   34
       yes  569  214

               Accuracy : 0.9399
                 95% CI : (0.9351, 0.9445)
    No Information Rate : 0.9753
    P-Value [Acc > NIR] : 1

                  Kappa : 0.3923

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.9419
            Specificity : 0.8629
         Pos Pred Value : 0.9963
         Neg Pred Value : 0.2733
             Prevalence : 0.9753
         Detection Rate : 0.9186
   Detection Prevalence : 0.9220
      Balanced Accuracy : 0.9024

       'Positive' Class : no
```

(iii) 결론

선형판별분석과 이차판별분석 중 더 "제주시 퇴근시간 승차인원이 많은 정류소"를 선택하는 데 적합한 모델은 선형판별분석이라 생각합니다. 그 이유는 퇴근시간 승차인원이 많은 곳을 미리 알고 대처를 하는 것이 목적일 때, 가능성이 높은 곳은 모두 고려 사항으로 두는 것이 좋을 것입니다. 따라서 Accuracy와 Sensitivity가 높은 선형판별분석을 선택합니다. 또한, 선형판별분석의 pos pred value와 neg pred value가 이차판별분석에 비해 더 높은 것도 확인 할 수 있습니다.

2.4. 서포트벡터분류기

(i) 선정 이유

로지스틱 회귀와 유사한 서포트벡터 분류기에 대해 적합하려 합니다. 손실함수의 유사성으로 인해 로지스틱 회귀와 서포트 벡터 분류기는 비슷한 결과를 준다고 알려져 있습니다. 이 때, 클래스들이 잘 분리되어 있을 때는 SVM이 로지스틱 회귀보다 더 나은 경향이 있으며, 좀 겹치는 경우에는 로지스틱 회귀가 더 선호됩니다.

데이터 탐색을 통해 클래스가 잘 분리되어 있지 않다는 것을 알고 있습니다. 하지만, 실제로 적합을 통해 어느 모델이 더 나은지 판단하겠습니다.

(서포트벡터머신으로 비선형 경계를 이용해 모델링을 할 수 있으나 앞서 선형·이차판별분석에서 비선형경계보다 선형경계가 더 나은 것을 확인했으므로 서포트벡터머신을 이용하지 않았습니다. 그리고 데이터 탐색에서 서포트벡터머신이 사용될 만큼 비선형적이지 않았습니다.)

(ii) 적합 및 예측

Cost option은 마진 위반에 대한 비용을 지정하며 서포트 벡터들만이 분류기에 영향을 주어 편향-분산 절충을 제어하는 조율 파라미터입니다.=. Cost 인자가 작으면 마진이 넓고, 많은 서포트 벡터들이 마진 상에 있거나 마진을 위반할 것입니다. 반대로, Cost 인자가 큰 경우 마진이 좁을

것이고 마진 상에 있거나 마진을 위반하는 서포트 벡터들이 소수가 될 것입니다.

따라서 cost 인자를 결정하는 것은 중요한 일이므로 0.001, 0.01, 0.1, 1, 5 이렇게 5가지를 주고 교차검증 방식을 사용하여 가장 좋은 결과를 주는 cost 인자를 구하였습니다. 그 결과 cost=0.01임을 알 수 있습니다.

Train set에 대해서 cost=0.01로 적합했고, validation set에 대해 예측한 결과 Accuracy=0.9856, Sensitivity=0.9968, Specificity=0.5403로 높게 나오는 것을 볼 수 있습니다.

```
Parameter tuning of 'svm':

- sampling method: 10-fold cross validation

- best parameters:
 cost
  0.1

- best performance: 0.01353404

- Detailed performance results:
   cost      error  dispersion
1 0.001 0.01392245 0.001911103
2 0.010 0.01362368 0.001905825
3 0.100 0.01353404 0.002045614
4 1.000 0.01356392 0.001997292
5 5.000 0.01356392 0.001997292
```

```
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  9762  114
       yes   31  134

               Accuracy : 0.9856
                 95% CI : (0.983, 0.9878)
    No Information Rate : 0.9753
    P-Value [Acc > NIR] : 5.288e-13

                  Kappa : 0.6418

 Mcnemar's Test P-Value : 9.778e-12

            Sensitivity : 0.9968
            Specificity : 0.5403
         Pos Pred Value : 0.9885
         Neg Pred Value : 0.8121
             Prevalence : 0.9753
         Detection Rate : 0.9722
   Detection Prevalence : 0.9836
      Balanced Accuracy : 0.7686

       'Positive' Class : no
```

2.5. Tree 기반 모델

로지스틱회귀, 선형·이차판별분석, SVC(서포트벡터분류기)와 다른 구조의 모델링을 했을 때, 더 좋은 결과가 나올 수도 있습니다. 그 중 Tree 구조를 선택했습니다.

(1) Classification Tree

(i) 선정 이유

분류트리 1개는 데이터에 대해 설명하기 쉽지만 예측 정확도가 떨어집니다. 따라서 Classification tree를 통해 예측력보다는 "퇴근시간 버스 승차인원이 많은 정류소"에 대한 설명변수들의 특징을 찾기 위해 사용하였습니다.

(ii) 적합 및 예측

Rpart 함수를 이용해 train set에 대해서 classification tree를 적합했습니다. (지니계수를 사용하는 함수입니다.) 그리고 validation set에 대해 예측하였고, Accuracy=0.9863, Sensitivity=0.9889, Specificity=0.8313으로 높게 나오는 것을 볼 수 있습니다. 과적합의 우려가 있어 교차검증을 한

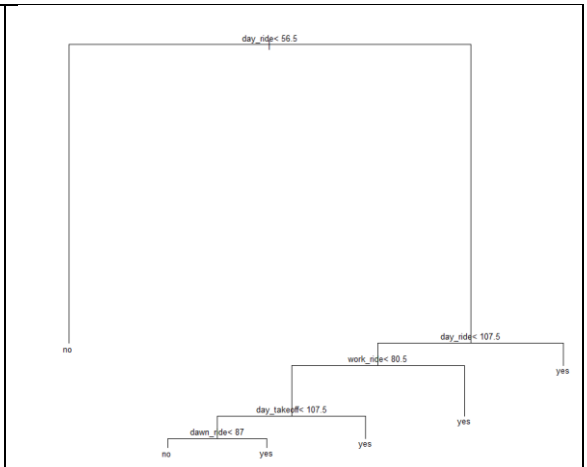
결과 xerror(교차검증 오류율)이 가장 최소가 될 때가 앞서 적합했던 모델임을 알 수 있었습니다. (rpart 내장 함수를 이용했습니다.)

(iii) Classification tree를 이용한 데이터 분석

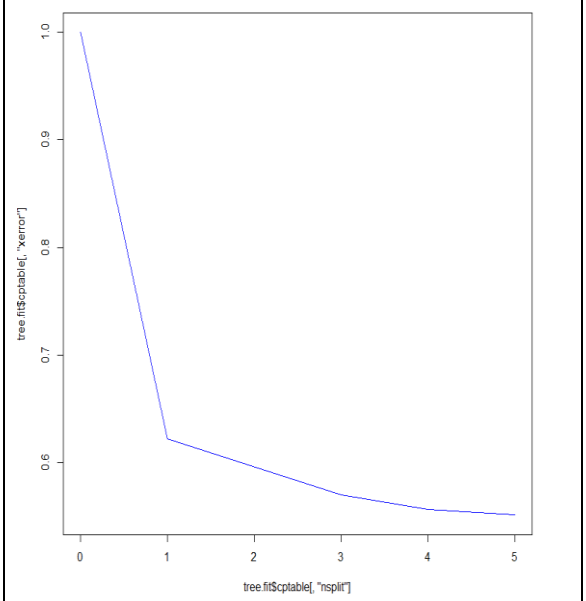
각 node에서 설명변수들의 수치를 봤을 때, 각 설명변수마다 일정 수치를 넘겨야 "퇴근시간 승차인원이 많은 정류소"로 선택이 됩니다. 즉, 기본적으로 이 정류소들은 퇴근시간 이외에도 다른 시간대에 사용하는 사람들이 많은, 유동인구가 어느 정도 있는 정류소들이라는 것을 확인시켜 줍니다.

로지스틱회귀모델에서 유의했던 설명변수 day_ride, work_ride, dawn_ride가 중복해서 사용된 것을 볼 수 있습니다. 로지스틱회귀모델에서 유의하지 않았던 Dawn_ride 변수가 추가로 발견이 되는데 바로 위에서 설명했던 패턴을 거스르지 않기 때문에 특이한 점은 발견할 수 없습니다

```
> tree.fit
n= 23430

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 23430 579 no (0.975288092 0.024711908)
   2) day_ride< 56.5 22866 185 no (0.991909385 0.008090615) *
   3) day_ride>=56.5 564 170 yes (0.301418440 0.698581560)
     6) day_ride< 107.5 383 166 yes (0.433420366 0.566579634)
      12) work_ride< 80.5 200  81 no (0.595000000 0.405000000)
        24) day_takeoff< 107.5 182  66 no (0.637362637 0.362637363)
          48) dawn_ride< 87 175  59 no (0.662857143 0.337142857) *
          49) dawn_ride>=87 7   0 yes (0.000000000 1.000000000) *
        25) day_takeoff>=107.5 18   3 yes (0.166666667 0.833333333) *
      13) work_ride>=80.5 183  47 yes (0.256830601 0.743169399) *
     7) day_ride>=107.5 181   4 yes (0.022099448 0.977900552) *
>
```




```
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  9765   28
       yes  110  138

               Accuracy : 0.9863
                 95% CI : (0.9838, 0.9884)
    No Information Rate : 0.9835
    P-Value [Acc > NIR] : 0.01383

                  Kappa : 0.6599

 Mcnemar's Test P-Value : 5.38e-12

            Sensitivity : 0.9889
            Specificity : 0.8313
         Pos Pred Value : 0.9971
         Neg Pred Value : 0.5565
             Prevalence : 0.9835
         Detection Rate : 0.9725
   Detection Prevalence : 0.9753
      Balanced Accuracy : 0.9101

       'Positive' Class : no
```



(2) Random Forest

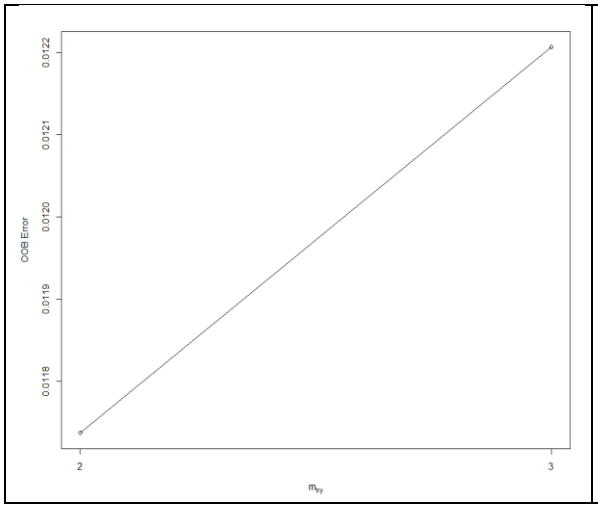
(i) 선정 이유

앞서 회귀분석에서와 마찬가지로 단일 Classification tree는 변동성이 큽니다. 또한, Bagging은 트리들간의 상관성이 높다는 문제점이 있습니다. 따라서 Random Forest를 사용하여 정확도를 높이고, 변수의 중요도를 살펴보도록 하겠습니다.

(ii) 적합 및 예측

Mtry, 즉 추출할 변수의 개수를 정하기 위해 tuneRF 함수를 이용했습니다. Oob error가 mtry=3에서 가장 작으므로 mtry를 3으로 지정해주어 적합을 하도록 하겠습니다.

300개의 tree를 만들어 train set에 대해 적합해 주었고, validation set에 대한 예측 결과는 Accuracy=0.9877, Sensitivity=0.9962, Specificity=0.9492로 높게 나오는 것을 볼 수 있습니다.




```
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  9756   87
       yes   37  161

               Accuracy : 0.9877
                 95% CI : (0.9853, 0.9897)
    No Information Rate : 0.9753
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7157

 Mcnemar's Test P-Value : 1.081e-05

            Sensitivity : 0.9962
            Specificity : 0.6492
         Pos Pred Value : 0.9912
         Neg Pred Value : 0.8131
             Prevalence : 0.9753
         Detection Rate : 0.9716
   Detection Prevalence : 0.9803
      Balanced Accuracy : 0.8227

       'Positive' Class : no
```

(iii) 변수 중요도

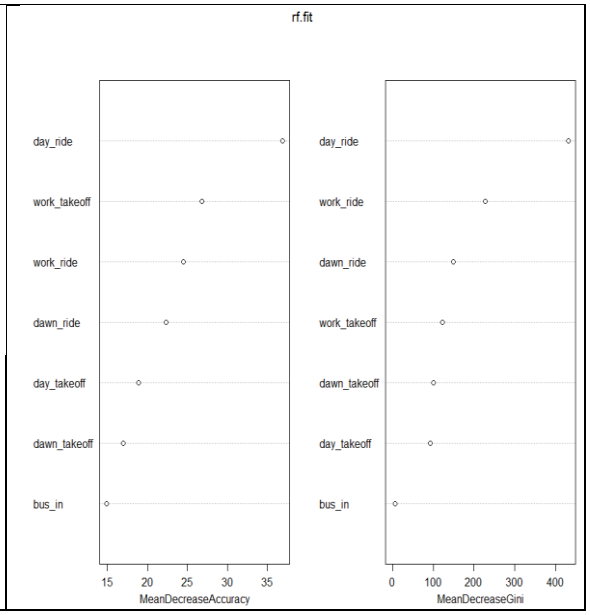
MeanDecreaseAccuracy는 해당 변수를 Permutation(순서섞기)한 이후 정확도의 감소를 의미합니다. MeanDecreaseGini는 해당 설명변수가 모델에 적용되는 것이 전체적인 분류의 불순도를 얼마나 감소시키는지를 의미합니다.

MeanDecreaseAccuracy에서는 day_ride, work_takeoff, work_ride, dawn_ride, day_takeoff, dawn_takeoff, dawn_takeoff 순으로 중요합니다.

MeanDecreaseGini는 day_ride, work_ride, dawn_ride, work_takeoff, dawn_takeoff, day_takeoff, bus_in 순으로 중요합니다.

두 가지 측도에서 공통적으로 있는 상위 Day_ride, work_ride, work_takeoff, dawn_ride 네 변수가 중요함을 알 수 있습니다.

```
> importance(rf.fit)
                   no       yes MeanDecreaseAccuracy MeanDecreaseGini
bus_in       7.450632 13.486856             14.86329        6.252535
dawn_ride   14.706315 18.281513             22.36266      149.837564
dawn_takeoff 8.813087 12.626140             16.99222      100.415306
work_ride   21.699541 12.377004             24.53968      228.034827
work_takeoff 24.869694  8.160626             26.84847      123.330985
day_ride    26.960848 31.642777             36.95306      431.476933
day_takeoff 17.123072  5.784319             18.85722       92.328461
```

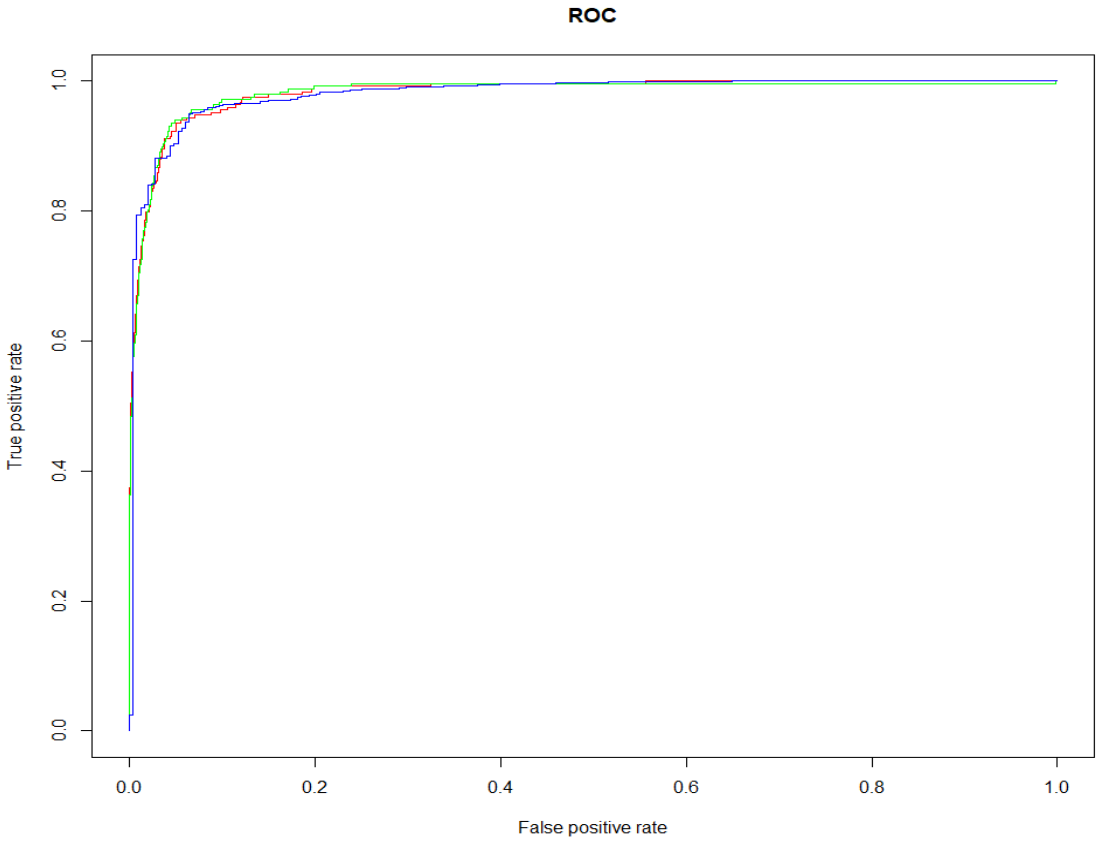


2.6. 최종 결과

(1) 적합 모델

로지스틱회귀모델, 선형·이차판별분석, 서포트벡터분류기, Classification Tree, Random Forest 총 7개의 모델을 적합하고 예측해 보았습니다. 아래 표는 Accuracy, Sensitivity, Specificity 결과들을 모두 요약한 표입니다. 또, Logistic(빨간색), LDA(초록색), SVC(파란색)의 ROC 곡선을 그린 그래프입니다.

	Accuracy	Sensitivity	Specificity
Logistic	0.9417	0.9418	0.9395
LDA	0.9834	0.9916	0.6573
QDA	0.9399	0.9419	0.8629
SVC	0.9856	0.9968	0.5403
Classification Tree	0.9863	0.9889	0.8313
Random Forest	0.9877	0.9962	0.9492



표와 ROC 곡선을 참고해서 가장 적합한 모델을 찾아보도록 하겠습니다.

앞서 LDA와 QDA 중 LDA를 분석 목적에 더 적합한 모델이라고 언급했습니다. 그 이유는 "퇴근 시간 승차인원이 많은 정류소"를 미리 알고 대처를 하는 것이 목적일 때, 가능성이 높은 곳은 모두 고려 사항으로 두는 것이 좋을 것이기 때문입니다. 따라서, Accuracy와 Sensitivity에 더 중요성을 두어 LDA를 선택했습니다.

LDA, 로지스틱, SVC를 비교해 보도록 하겠습니다. ROC 그래프를 보면 LDA, Logistic, QDA 모두 곡선 아래 비슷한 면적을 가지고 있습니다. 따라서 표를 통해 더 자세히 살펴보겠습니다. 각 로지스틱이 가장 Specificity가 높지만, Accuracy와 Sensitivity 면에서는 가장 낮습니다. 그 뒤로 SVC, LDA 순으로 Accuracy와 Sensitivity가 높고, Specificity가 낮습니다. LDA는 설명변수가 다변량정규분포를 따른다는 가정 하의 모델이므로 모델의 적합성이 의심되어 고려대상에서 제외하겠습니다.

Logistic와 SVC를 비교하면서 상기해야 할 것은 타겟 관측치가 전체 관측치 중 매우 낮아 2.5%를 차지한다는 것입니다. 따라서 아무것도 분류하지 않아도 오차율이 2.5%, 즉 정확도가 97.5%가 된다는 것입니다. Accuracy와 Sensitivity에 더 중요성을 둔다는 점도 감안했을 때 Accuracy와 Sensitivity가 높은 SVC를 선택하는 것이 좋다고 생각합니다. SVC는 개별 관측치에 robust한 특징

이 있기 때문에 매우 많은 승차인원을 가진 정류소의 경우 큰 오류 없이 발견할 수 있을 것을 기대할 수 있습니다.

SVC와 Random Forest를 비교해 보면 Accuracy, Sensitivity, Specificity 모두 Random Forest가 높은 것을 알 수 있습니다. 약 300개의 트리를 이용해 결합하여 결과를 도출했으며 클래스들이 잘 분리되어 있는 상태가 아니기 때문에 Random Forest가 매우 적합하게 나왔다고 생각합니다.

Random Forest는 임의로 나누어준 train, test set에 대해서만 성능이 좋을 수 있습니다. 따라서 교차검증을 통해 Random Forest의 보편적인 성능을 알아보려 합니다. 또한, 과적합이 일어나는지도 확인해 볼 것입니다. 그 결과 5 fold 교차검증을 통해 나눈 각각의 5개의 평균이 Accuracy=0.996, sensitivity=0.8767, specificity=0.9969가 나왔습니다. 임의로 나누어준 data set 이외에도 sensitivity가 다소 떨어졌지만 성능이 좋은 것을 확인 할 수 있습니다.

```
> apply(mean.rmse, 1, mean)
[1] 0.9959070 0.8766849 0.9968820
```

따라서 7개의 분류 모델 중 Random Forest가 "제주도 퇴근시간 많은 승차인원을 가진 정류소"를 예측하는 데 가장 적합한 모델이라는 결론을 얻었습니다.

(2) 결과 분석

로지스틱회귀를 통해 얻어낸 획귀계수 추정치와 tree 모델을 통해 얻은 정보와 변수의 중요도를 종합적으로 참고하여 "퇴근시간 승차 인원이 많은 버스정류장"의 특징에 대해서 분석해 보도록 하겠습니다.

유의한계수	Dawn_ride	Work_ride	Day_ride	Dawn_takeoff	Bus_in
로지스틱	0.011688	0.01485	0.05797	0.023843	-1.332415
Randomforest	중요	중요	중요	중요	안중요함

로지스틱회귀계수 추정치들 중 유의한 변수들과 random forest 모델에서 중요한 변수로 나타났던 변수들에 대해서 각각 어떤 영향을 주는지 알아보도록 하겠습니다. 특히, 두 모델에서 공통적으로 dawn_ride, day_ride, work_ride 가 가장 유의하며 중요하게 나온 것을 통해 세 변수가 "퇴근시간 승차인원이 많은 버스정류장"에 가장 큰 영향력을 주는 것을 알 수 있습니다.

Day_ride, work_ride, dawn_ride, dawn_takeoff의 경우 계수추정치가 양수이며 이 시간대에 승차인원이 많을수록 "퇴근시간 승차인원이 많은 정류소"가 될 확률이 높습니다.

승차 관련 변수들의 계수 추정치를 자세히 들여다보면 dawn_ride=0.011688, work_ride=0.01485, day_ride=0.05797로 점차 중요성이 커지는 것을 알 수 있습니다. 데이터 탐색에서 "퇴근시간 승차인원이 많은 정류소"의 경우 시간이 지남에 따라 유동인구가 많아진다고 했으므로 "그렇지 않은 정류소"보다 유동인구의 변화량이 확연히 차이나는 순으로 중요성이 커졌다고 볼 수 있습니다.

Dawn_takeoff는 시간대가 6시~8시입니다. 이 시간대에 "그렇지 않은 정류소", 즉 번화가가 아닌 곳에서는 버스를 이용해 그곳에 도착하려는 사람들이 매우 적을 것입니다. 하지만, "퇴근시간 승차인원이 많은 정류소"는 도심, 또는 회사에 위치해 있을 것이라 추론했고, 그에 따라 이른 시간임에도 불구하고 "그렇지 않은 정류소"에 비해 도착하려는 사람들이 많을 것입니다.

이와 비슷한 맥락으로 "그렇지 않은 정류소"에 비해 "퇴근시간 승차인원이 많은 정류소"는 시간대에 상관없이 더 많은 사람들이 기본적으로 이용할 것입니다. 이는 classification 모델에서 각 node의 기준들이 위와 같은 경향을 띄고 있는 것을 통해 확인할 수 있었습니다.

한 가지 해석하기 어려운 것은 bus_in 변수입니다. 시내버스와 시외버스를 구분하는 변수로 로지스틱회귀에서는 유의하면서도 중요한 변수로 나왔지만 tree 모델에서는 가장 중요하지 않은 변수로 고려되었습니다. 소수의 정류장에 해당이 되거나 이상치로 인해 과대하게 부풀려진 것은 아닌지 생각했습니다. 그럼에도 만약 bus_in이 "퇴근시간 승차인원이 많은 정류장"과 연관된 패턴을 가진다면, 제주도의 특성 때문이라 생각했습니다. 제주도는 시가 2개입니다. 그렇기 때문에 한정된 번화지역에서 이동할 때 시내버스와 시외 버스를 모두 이용한다 했을 때, 퇴근시간에 보통 주거지역으로 이동하면서 시외버스 이용이 많아질 것입니다. 하지만 애매한 점이 있기 때문에 "퇴근시간 승차인원이 많은 버스 정류소"의 특징으로서 선택하지 않을 것입니다.

Ⅲ. 결론

1. 최종 결과 요약

전체적인 퇴근시간 승차인원을 파악 및 예측하는 데 day_ride, work_ride work_takeoff, dawn_ride 변수가 다른 변수들에 비해 중요함을 알 수 있었습니다. Day_ride, work_ride가 클수록 전체적인 퇴근시간 승차인원이 뚜렷하게 커지는 것을 알 수 있었습니다. work_takeoff는 비록 이상치의 영향을 받지만 이 역시 클수록 전체적인 퇴근시간 승차인원이 많아졌습니다. 새벽시간 승차인원은 적을수록 퇴근시간 승차인원이 많아지는 것을 알 수 있었지만 매우 많은 퇴근시간 인원을 가지는 정류소에서 일어나는 특정 현상이었습니다.

(dawn_takeoff, day_takeoff의 경우 해석의 주가 되었던 선형성 모델의 한계와 퇴근시간 인원이 많은 정류소와 퇴근시간 인원이 적은 정류소의 특징이 섞인 것 때문에 해석이 어려웠습니다.)

퇴근시간 승차인원이 매우 많은 정류소의 특징을 파악 및 예측하는 데 dawn_ride, day_ride, work_ride가 다른 변수들에 비해 중요함을 알 수 있었습니다. 위의 전체적인 흐름과 마찬가지로 시간과 관계없이 승차, 하차 인원이 많을수록 "퇴근시간 승차인원이 많은 정류소"일 확률이 높았습니다. 특히, day_ride, work_ride가 뚜렷이 많을수록 확률이 높아졌고, dawn_takeoff도 이러한 흐름의 영향을 받아 하차인원이 많을수록 확률이 높아집니다.

(bus_in 변수는 로지스틱에서는 유의하게 나왔지만, tree 모델에서 중요도가 낮은 변수였습니다. 로지스틱에서 적합이 될 때, 이상치 또는 다른 문제로 인해 과대해석되었다고 생각합니다.)

결과를 통해 전체 정류소의 퇴근시간 승차인원과 퇴근시간 승차인원이 매우 많은 정류소들은 새벽, 출근, 낮 시간의 승차, 하차 인구 수가 많을수록 퇴근시간 승차인원이 많아지는 공통점을 가지고 있습니다. 다만, 퇴근시간 승차인원이 매우 많은 정류소가 대체로 "그렇지 않은 정류소"보다 각 설명변수에서 수치가 대체적으로 더 높았으며 출근시간 하차 인원 대비 퇴근시간 승차 인원이 매우 높았습니다. 또한, 매우 많은 퇴근시간 승차인원이 있는 곳은 오히려 새벽시간 승차인원이 비슷한 범주에서는 더 적다는 특징도 알 수 있었습니다.

2. 활용 방안

제주도내 주민등록인구는 2019 년 11 월 기준 69 만명으로, 연평균 4%대로 성장했습니다. 외국인과 관광객까지 고려하면 전체 상주인구는 90 만명을 넘을 것으로 추정됩니다. 제주도민 증가와 외국인의 증가로 현재 제주도의 교통체증이 심각한 문제로 떠오르고 있습니다. 이 문제를 해결하는 방법으로 예측모델을 사용해 미리 퇴근시간 승차인원이 많은 곳의 정류소와 그 수를 알고 대처하는 것입니다.

하지만, 미리 대비하기 위해서는 하루동안의 버스정류소와 관련된 정보들을 가진 데이터가 준비되어 있어야 합니다. 실현가능성이 없어 보이지만 꼭 그렇지만도 않습니다.

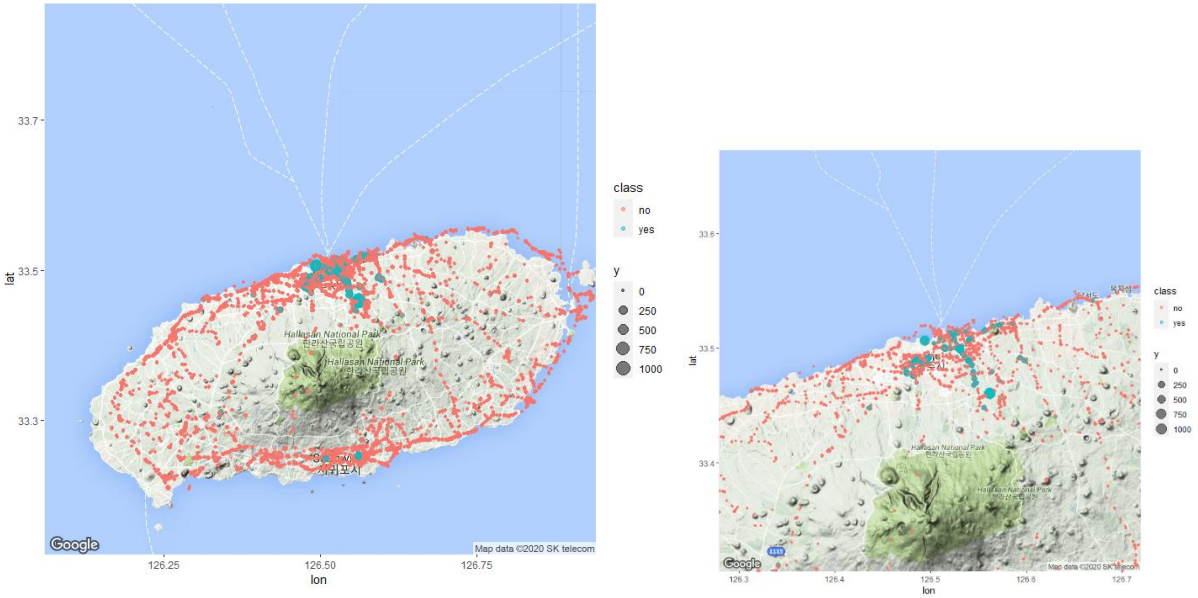
오늘날에는 모든 정보가 신속히 모입니다. 특히, 버스의 경우 승차할 때와 하차할 때 버스 카드를 찍어야 합니다. 카드를 찍었을 때 누가, 어디서, 어떤 버스를, 어디서 내렸는지 등 모든 정보가 카드 회사, 버스 회사 등에 바로 전달이 되고 축적이 됩니다. 뿐만 아니라 정보는 충분히 신뢰성이 있습니다. 교통청에서 실험한 결과 교통카드를 이용한 승차인원과 실제 대중교통 승차 인원과 비교한 결과 교통카드를 이용한 승차인원이 약 5% 정도 높았다고 합니다. 따라서 오차가 있겠지만 꽤 높은 정확성으로 얼마나 버스에서 승차했는지, 하차했는지에 대한 정보를 모을 수 있습니다.

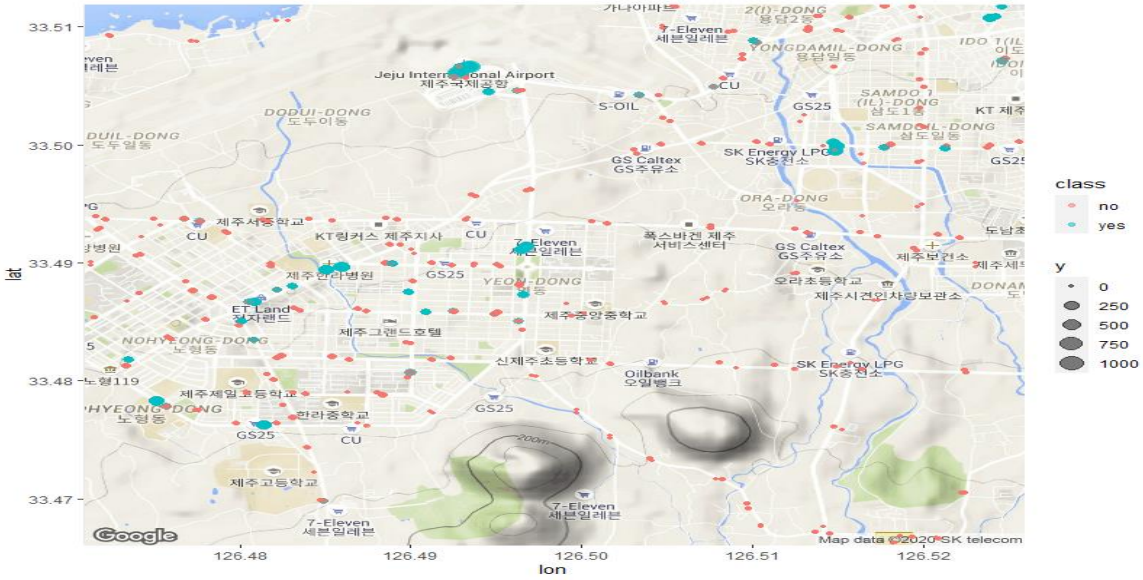
신속하고 정확한 데이터를 기반으로 이 보고서에서 적합한 Random Forest 회귀모델과 Random Forest 분류모델을 이용할 수 있습니다. 분류모델을 이용해 어느 정류소가 퇴근시간 승차인원이 기준보다 많을 것인지를 예측할 수 있습니다. 이를 통해 대략적으로 그 날 하루동안 교통혼잡이 일어날 수 있는 대략적인 가능성을 추측해 볼 수 있을 것입니다. 또한, 회귀모델을 이용해 구체적으로 예상되는 인원을 알고, 교통혼잡에 대응할 방식을 세우는 데 도움이 될 것입니다.

그리고 버스정류소를 기준으로 묶은 데이터이기 때문에 정류소를 지도에 표시할 수도 있습니다. 두 모델을 사용해 예측한 결과들을 지도에 시각적으로 표현해 좀 더 편하게 교통혼잡이 일어날 지역을 파악하고 교통혼잡의 수준도 알 수 있습니다.

예시로, 적합한 random forest 회귀 모델과 random forest 분류 모델을 이용해 분석에 사용했던 데이터의 버스승차 인원을 예측하고, 탑승인원이 60명이 넘는 곳을 지도에 표시해 보도록 하겠습니다.

원의 크기가 클수록 그 정류소에 승차인원이 많은 것을 의미하며, 원의 색깔이 민트색인 경우 탑승인원이 60명이 넘습니다. 더 자세히 보고 싶을 때는 확대해서 그 주변의 어떤 정류소들이 있고, 주변에 무슨 건물들이 있는지 등의 교통혼잡을 예방에 도움이 되는 추가적인 정보도 얻을 수 있을 것입니다.





이 과정들은 모두 교통청이나 버스 회사에서 이용할 수 있으며 제주도에서의 심각한 교통체증을 빠르게 대처할 수 있도록 해주는 좋은 참고자료가 될 것입니다