다중회귀분석을 통한 날씨에 따른 극장 관객 수에 관한 연구

20176735 김수린

목차

1. 연구 소개

1) 주제 선정 배경

영화는 한국인의 여가 생활에서 빼놓을 수 없는 존재이다. 영화진흥위원회(KOFIC)의 2019년 한국 영화산업 결산 보고서에 따르면 “2019년 전체 극장 관객 수는 2억 2668만 명으로 전년 대비 4.8% 증가” 하여 역대 최고 관객 수를 기록하였으며 인구 1인당 연평균 관람 횟수는 4.37회로 아이슬란드의 수치인 4.32회를 넘는 세계 최고 수준이다. 하지만 영화 산업의 전체 매출 중 디지털 온라인과 해외 매출을 제외한 극장 매출이 차지하는 비중이 2019년까지 항상 75% 이상으로 높은 수치를 유지하고 있음과 다른 여러 상황들을 종합해보았을 때, 한국은 ‘영화광들의 나라’보다는 ‘극장을 많이 찾는 나라’에 가까우며 야외 여가 중 싸고 오래 즐길 수 있는 활동으로 영화 관람을 선택한 것으로 분석된다. (이문원, 2020)

본 연구는 이처럼 사람들이 상영 중인 영화에 대한 선호에 따라 극장 영화관람을 선택하는 것보다 수많은 야외 여가 중 하나로서 이를 선택하는 것에 관심을 가져 시작되었고 특히, 그 이유 중에서도 개인적인 경험을 고려하여 날씨에 주목하였다. 또한, 영화 흥행 예측에 대한 연구들은 이미 많이 존재하지만 날씨와의 관계를 밝힌 연구는 수가 적어 본 연구를 통해 직접 확인해보고자 하였으며, 만약 그 유의미한 관계가 존재한다면 영화개봉 일자를 보다 효율적으로 선정할 수 있을 것이라 기대하였다. 한편, 관련한 선행 연구에 따르면 여름 관객은 날씨가 뜨겁고 불쾌지수가 높을수록

영화관을 더 방문했다. (김형호, 2016) 따라서 본 연구는 선행연구와 차이점을 두기 위해 여름에서 1년으로 연구 범위를 확장하여 2019년 한 해간 특정 날씨 상황에 따라 영화 관객 수에 유의미한 차이가 존재하는지 회귀분석을 통한 분석을 진행하였다.

2) 데이터 수집 및 정제

본 연구의 대상은 COVID-19 이전인 2019년 1월 1일부터 2019년 12월 31일까지의 일일 데이터이다. 종속 변수는 영화진흥위원회(KOFIC) 박스오피스에서 제공하는 전국 일별 총관객 수를 사용하였으며, 독립 변수인 날씨 데이터는 기상자료 개방포털에서 제공하는 서울지점(지점번호; 108)의 종관기상관측(ASOS)과 황사관측(PM10)자료들 중 평균기온, 최저기온, 최고기온, 일 강수량, 최대순간풍속, 평균 풍속, 평균 상대습도, 합계 일조시간, 합계 일사량, 1시간 최대 일사량, 일 최심신적설, 평균전운량, 1.5m 지중온도, 일 미세먼지 농도 데이터를 수집하여 최종적으로는 이 중 일부를 활용하였다.

독립 변수와 종속 변수 데이터의 수집 범위가 다른 것은 서울의 일일 총관객 수 데이터를 얻는 것에 어려움이 있었기 때문이다. 더불어, 서울의 관객 수 비중이 비교적 일정하여 영화시장 분석가 김형호(2016)의 연구에서도 서울 대신 전국의 관객 수로 분석을 진행한 바가 있어 본 연구도 관객 수는 전국 기준을 사용하였다.

이제 본격적으로 분석을 시작하기에 앞서 데이터를 정제하고 불필요한 변수들을 먼저 제거하였다. 결측치가 많은 경우 분석의 정확성을 위해 해당 변수 자체를 삭제하였는데, 일 강수량, 최심신적설, 일 미세먼지 농도 데이터가 각각 결측치가 226개, 359개, 27개로 상당수 존재하였으므로 제거되었다. 이 중 특히, 일 강수량의 경우에는 평균 상대습도와의 상관분석 결과 유의한 상관관계가 존재하므로(r = .55, p < .001) 제거하여도 정보의 손실이 많지 않을 것이라 예상하였다. 결측치가 한두 개로 적게 나타난 합계 일조시간, 합계 일사량, 1시간 최대 일사량, 1.5m 지중온도의 경우 해당 결측치를 갖고 있는 240, 241, 284번째 행을 삭제하여 데이터에서 결측치를 모두 제거하였다.

다음으로 모든 독립 변수들 간의 상관관계를 확인한 <표 1>을 보면 서로 높은 상관관계를 가지는 변수들이 세 그룹으로 나타난다. 이러한 결과는 해당 그룹 내의 변수들이 각각 대기의 온도, 바람의 속도, 햇빛의 양에 대한 정보를 공통적으로 표현하고 있어 나타났을 것으로 추측할 수 있다. 따라서 세 그룹에서 각각 하나의 변수만 선택하기 위해 각각의 독립 변수들로 단순 회귀분석을 실시하였고 <표 2>의 결과를 토대로 각 그룹에서 가장 유의 확률이 낮은 변수 하나만을 선택하였다. 그 결과 최종적으로 선택된 독립 변수는 다음과 같다. (모두 하루 기준이며 괄호 안은 R 코드 내에서 사용한 변수명을 의미함)

- 관객 수(aud) : 전국 총관객 수 (천 명)
- 최저기온(lowtem) : 가장 낮은 기온 (℃)

- 최대순간풍속(maxWS) : 순간 풍속의 최댓값 (m/s)
- 평균 상대습도(avgHum) : 포화 수증기량에 대한 현재 수증기량 비율의 평균 (%)
- 1시간 최다 일사량(maxSun) : 1시간 동안 태양에서 지구로 오는 에너지가 단위 면적당 닿는 양의 최댓값 (MJ/m2)
- 1.5m 지중온도(tem15) : 지면으로부터 1.5m 깊이에서의 토양의 온도 (℃)
- 평균전운량(avgCl) : 하늘 전체에 대해 구름이 덮은 양의 평균 (0~10)

표 1. 독립 변수들 간의 상관관계

변수	1	2	3	4	5	6	7	8	9	10
1 평균기온	–									
2 최저기온	**.99*****	–								
3 최고기온	**.99*****	**.96*****	–							
4 평균풍속	−.15**	−.12*	−.17***	–						
5 최대순간풍속	−.17**	−.16**	−.17***	**.83*****	–					
6 평균상대습도	.41***	.48***	.33***	−.08	−.14**	–				
7 합계일조시간	.04	−.04	.15**	.04	.12*	−.60***	–			
8 합계일사량	.40***	.31***	.49***	−.02	.04	−.44***	**.85*****	–		
9 1시간 최다일사량	.35***	.27***	.45***	−.01	.05	−.44***	**.84*****	**.96*****	–	
10 1.5m 지중온도	.81***	.82***	.77***	−.01	−.13*	.49***	−.04	.12*	.12*	–
11. 평균전운량	.31***	.38***	.22***	−.07	−.16**	.60***	−.80***	−.52***	−.52***	.25***

주. N = 243.
$^{*}p < .05$. $^{**}p < .01$. $^{***}p < .001$

표 2. 각 독립 변수에 따른 관객 수 단순 회귀분석 (단, 종속 변수는 $\sqrt{\text{관객수}}$)

독립변수	*df*	$\hat{\beta}$	*SE*	*t*	*p*	*adj* R^2
평균기온	360	0.04	0.04	1.02	.308	< .001
최저기온		**0.05**	**0.04**	**1.29**	**.198**	**.002**
최고기온		0.03	0.04	0.70	.487	< .001
평균풍속	360	−1.43	0.66	−2.15	.032	.001
최대순간풍속		**−1.05**	**0.37**	**−2.83**	**.005**	**.020**
합계일조시간	360	−0.13	0.10	−1.34	.180	.002
합계일사량		−0.07	0.06	−1.23	.220	.001
1시간 최다일사량		**−0.82**	**0.05**	**−1.63**	**.104**	**.005**

2. 다중선형회귀분석

본 연구의 경우 독립 변수가 두 개 이상이고, 독립 변수와 종속 변수 간의 선형 관계를 가정하여 다중선형회귀분석을 실시하였다. 특히, 완전 모형에서부터 시작하여 회귀분석의 기본 가정들이 성립하는지 단계적으로 확인하고 그에 따라 모형을 보완해나

가면서 최종 모형을 찾는다. 다중선형회귀분석의 일반적인 식과 기본 가정은 다음과 같다.

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip-1} + \epsilon_i, \quad i = 1, \cdots, n \ (Y = X\beta + \epsilon)$$

- $E(\epsilon_i) = 0, \quad i = 1, \cdots, n$
- $\mathrm{var}(\epsilon_i) = \sigma^2, \quad i = 1, \cdots, n$
- $\epsilon_1, \epsilon_2, \cdots, \epsilon_n$: 독립
- rank(X) = p
- $\epsilon_i \sim N(0, \sigma^2)$

1) 완전 모형(Full model, 모형 1)

$$\text{관객수} = \beta_0 + \beta_1 \text{기온} + \beta_2 \text{풍속} + \beta_3 \text{습도} + \beta_4 \text{일사량} + \beta_5 \text{지중온도} + \beta_6 \text{운량} + \epsilon \quad (1)$$

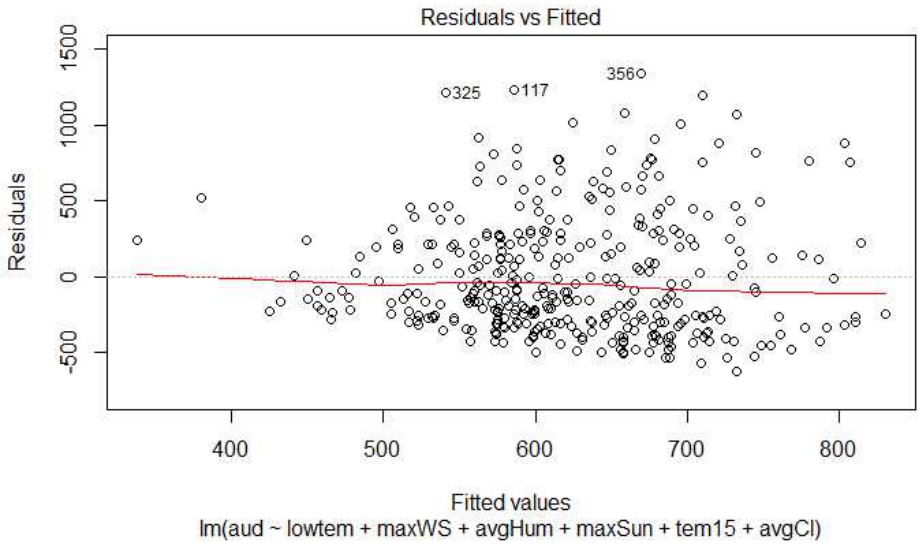


그림 1. 모형 1의 $(\hat{y},\ r)$ 그래프

첫 번째 모형은 6개의 모든 독립 변수들을 포함한 모형으로 식 (1)과 같이 나타낼 수 있다. 하지만 <그림 1>을 살펴보면 $\hat{y}$값이 커질수록 잔차의 분산이 커지는 것처럼 보이며 즉, 오차항 분산의 동질성 가정을 만족하지 않는 것처럼 보인다. 이는 종속변수가 '관객 수'에 대한 데이터이고, 따라서 포아송분포를 따른다고 본다면 자연스러운 결과이다. 포아송분포를 따르는 변수의 분산은 평균과 같기 때문이다. 따라서 분산을 동질적으로 만들기 위해 종속 변수에 루트를 취하여 다음 모형을 진행하였다.

2) 변수 변환 모형 (모형 2)

$$\sqrt{\text{관객수}} = \beta_0 + \beta_1 \text{기온} + \beta_2 \text{풍속} + \beta_3 \text{습도} + \beta_4 \text{일사량} + \beta_5 \text{지중온도} + \beta_6 \text{운량} + \epsilon \quad (2)$$

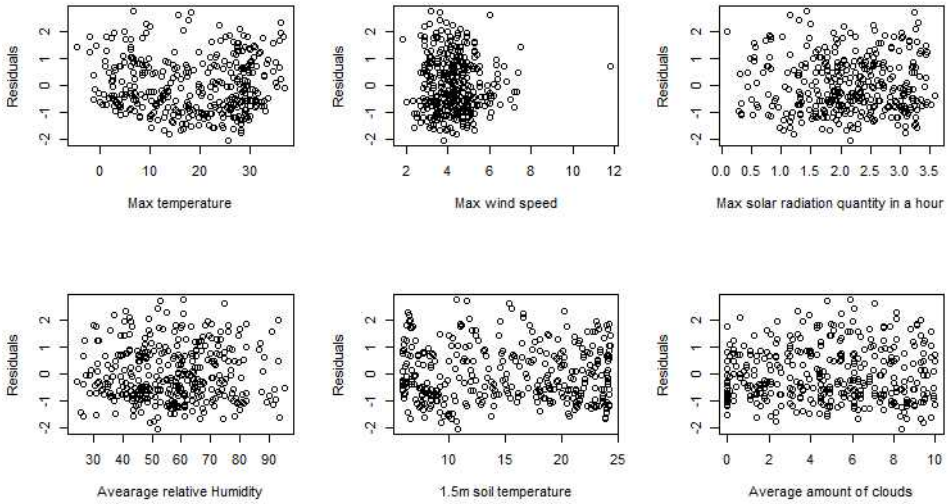


그림 2. 모형 2의 (x, r) 그래프

두 번째로, 식 (2)로 표현되는 관객 수에 루트를 취한 모형의 경우 <그림 2>와 ($\hat{y}$, r) 그래프를 그려보았을 때, 잔차들이 이전보다 고르게 분포된 것처럼 보이며, 분산의 동질성 가정을 어느 정도 만족하는 것처럼 보인다.

그리하여 다음으로 이상치 유무를 확인해본 결과 34, 69, 96, 133, 248, 263, 319, 361, 362 총 9개의 점이 $p_{ii} > 2\frac{p}{n}$를 만족하는 leverage point로 확인되었으며, 이 점들이 영향점에도 해당되는지 확인하기 위해 Cook's distance와 Difference in Fits (DFITS)를 활용하였다. 그 결과, Cook's distance의 기준($> F(p, n-p; 0.5)$)을 만족하는 영향점은 없었으나 DFITS를 기준($> 2\sqrt{\frac{p}{n-p}}$)으로 하면 총 17개의 점이 영향점으로 나타났다. 하지만 이 영향점들의 경우 잘못 측정되었거나 오류라고 보기는 어렵기 때문에 이들을 모두 제거하기보다는 <그림 3>을 통해 다른 점들과 확연한 차이를 보이는 248번째 값만을 제거하기로 하였다.

그리하여 해당 값을 제거한 후, 똑같이 모형을 세우고**(모형 3)** 여전히 분산의 동질성 가정이 성립함과 새로운 이상치가 존재하지 않음을 확인하였으며 다음 단계로 자기상관이 존재하는지에 대한 두 가지 검정을 실시하였다. 두 가지 검정의 귀무가설과 대립가설은 다음과 같다.

$$H_0 : \rho = 0$$
$$H_1 : \rho > 0$$

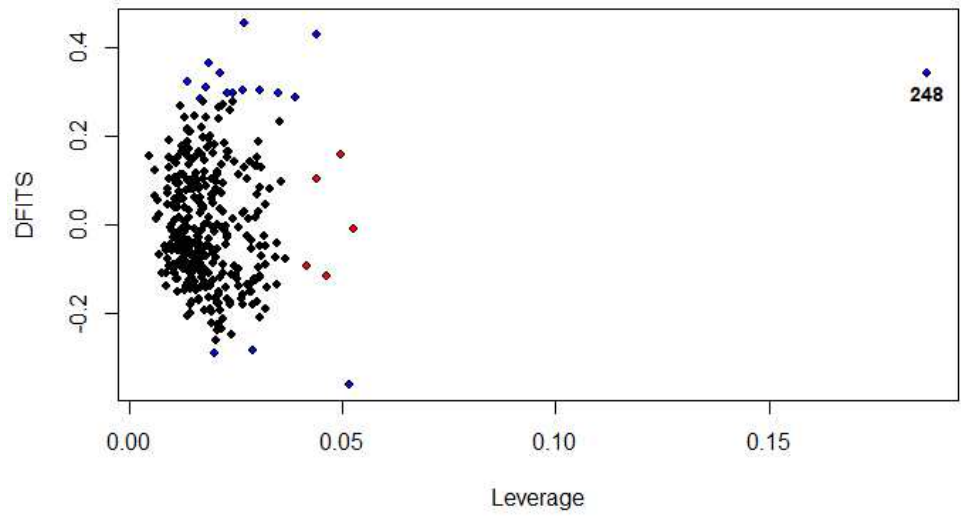


그림 3. 모형 2의 leverage vs dfits (빨강: high leverage 값, 파랑: 영향점)

더빈-왓슨 검정 결과 *d-value* = 0.92, $p < .001$ 로 귀무가설을 기각할 수 있으며, runs 검정 결과 또한 *z-value* = -7.60, $p < .001$ 로 귀무가설을 기각할 수 있다. 따라서 오차항들 사이에는 강한 양의 자기상관이 존재하며, 이는 오차항의 독립성 가정이 위배되었음을 의미한다. 본 연구에서는 이러한 자기상관을 해결하고자 두 가지 방법을 시도하였다. lag 1 표본자기상관계수가 .532, lag 7 표본자기상관계수가 .486으로 높게 나타났기 때문에 lag 1 자기상관과 관련하여 AR(1) 모형을, lag 7 자기상관과 관련하여 요일변수를 추가한 모형을 전개하였다.

3) AR(1) 모형 (모형 4)

$$\sqrt{Y}^{*} = \beta_0^{*} + \beta_1^{*}X_1^{*} + \beta_2^{*}X_2^{*} + \beta_3^{*}X_3^{*} + \beta_4^{*}X_4^{*} + \beta_5^{*}X_5^{*} + \beta_6^{*}X_6^{*} + \epsilon^{*} \quad (3)$$

AR(1) 모형은 lag 1 자기상관을 해결하는 한 가지 방법으로, 모든 종속, 독립 변수들을 $X_{t,i}^{*} = X_{t,i} - \hat{\rho}X_{t-1,i}$ 이와 같은 방식으로 변환하여 새롭게 모형을 세우며, 식 (3)으로 나타낸다. 여기서 $\hat{\rho}$는 lag 1 표본자기상관계수로 .532를 사용하였다. 분석 결과 5% 유의수준 하에서의 t-test에서 유의한 회귀계수가 상수항을 제외하고 하나도 없었으며 $H_0 : \beta_1 = \beta_2 = \cdots = \beta_6 = 0$, $H_1 : \text{Not } H_0$ 를 가설로 하는 모델 전체에 대한 F-test의 결과 또한 f = 1.22, $\mathrm{P}(\mathrm{F}(6, 353) > 1.22) = .3$으로 나타나 5% 유의수준에서 귀무가설을 기각할 수 없었다. 한편, 자기상관에 대한 검정 ($H_0 : \rho = 0$ vs $H_1 : \rho > 0$) 중 더빈-왓슨 검정에서 *d-value* = 1.69, p = .002, runs 검정에서 *z-value* = -2.85, p = 0.002 라는 결과를 얻어 두 검정 모두 귀무가설을 기각할 수 있었다. 따

라서 모형 4는 유의하지 않을 뿐만 아니라 여전히 인접한 오차끼리 유의미한 자기상관이 존재하므로 더 이상 나아가지 않고 이전 모형인 모형 3에 새로운 변수를 추가하는 다음 모형을 전개하였다.

4) 요일 변수 추가 모형 (모형 5)

$$\sqrt{Y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_5 + \beta_6 X_6 + \gamma_1 \text{day2} + \cdots + \gamma_6 \text{day7} + \epsilon \quad (4)$$

모형 3에서 lag 7 표본자기상관계수가 높게 나왔기 때문에 요일과 관객 수가 관련이 있을 것으로 생각하고 요일 변수를 추가하였다. 요일은 월요일부터 일요일까지 1,2,...,7로 나타내는 범주형 변수이며 월요일이 reference가 되어 회귀식이 식 (4)처럼 나타난다. 회귀분석 결과, 이상치가 존재하지 않으며 분산의 동질성 가정을 만족하였다. 5% 유의수준 하에서 유의한 설명 변수는 1시간 최대 일사량(p = .03), day3(p < .001), day4(p = .001), day5(p < .001), day6(p < .001), day7(p < .001)로, 요일 변수 대부분이 매우 유의하게 나타났고, $H_0 : \beta_1 = \beta_2 = \cdots = \gamma_7 = 0$, 를 가$H_1 : \text{Not H}_0$설로 하는 모델 전체에 대한 F-test 또한 f = 30.6, $\text{P}(\text{F}(9, 351) > 30.6) < .001$ 로 나타나 5% 유의수준에서 귀무가설을 기각할 수 있으며 매우 유의한 모형임을 알 수 있었다.

5) 모형 선택 및 최종모형

$$\sqrt{\text{관객수}} = \beta_0 + \beta_1 \text{풍속} + \beta_2 \text{일사량} + \beta_3 \text{기온} + \gamma_1 \text{day2} + \cdots + \gamma_6 \text{day7} + \epsilon \quad (5)$$

모형 5를 완전 모형으로 하여 AIC를 기반으로 하는 전진선택법, 후진소거법, 단계적 선택법을 각각 진행하였다. 그 결과 최종적으로 나타난 모형은 세 가지 방법에서 모두 식 (5)과 같았으며, 평균상대습도, 1.5m 지중온도, 평균전운량 변수가 삭제되었다. 해당 모형의 결과를 살펴보면 기온(최저기온) 변수의 회귀계수에 대한 유의확률이 상대적으로 높은 편(p = .124)으로 나타나 해당 변수를 제거한 모형을 축소 모형으로 하여 분산분석을 실시하여 모형을 조금 더 축소하고자 하였다. 그 결과 <표 3>와 같이 귀무가설을 기각할 수 없었으며, 축소 모형을 선택하게 되었다.

다시 한 번 남은 변수들 중 유의확률이 가장 높은(p = .070) 풍속(최대순간풍속)변수를 제외한 모형을 축소모형으로 하여 분산분석을 실시한 결과, <표 4>에서 볼 수 있듯이 유의확률이 .07로 5% 유의수준 하에서는 기각할 수 없지만 1% 유의수준에서는 기각할 수 있는 수준으로 나타났고, 본 연구에서는 귀무가설을 기각하는 것으로 결정하고 최종적으로 $\sqrt{\text{관객수}} \sim \text{최대순간풍속} + \text{최대일사량} + \text{요일더미변수}$ 모형을 선택하였다.

표 3. 최저기온 변수를 제거한 모형에 대한 분산분석

	Res.Df	RSS	Df	SSE	*F*	*p*
1	352	11703				
2	351	11624	1	78.63	2.3744	.124

주. H_0 : $\sqrt{\text{관객수}} \sim$ 최대순간풍속 + 최대일사량 + 요일더미변수

vs. H_1 : $\sqrt{\text{관객수}} \sim$ 최대순간풍속 + 최대일사량 + 최저기온 + 요일더미변수

표 4. 최대순간풍속 변수를 제거한 모형에 대한 분산분석

	Res.Df	RSS	Df	SSE	*F*	*p*
1	353	11813				
2	352	11703	1	110.03	3.31	.070

주. H_0 : $\sqrt{\text{관객수}} \sim$ 최대일사량 + 요일더미변수

vs. H_1 : $\sqrt{\text{관객수}} \sim$ 최대순간풍속 + 최대일사량 + 요일더미변수

3. 최종모형 분석

$$\sqrt{\text{관객수}} = \beta_0 + \beta_1 \text{풍속} + \beta_2 \text{일사량} + \gamma_1 \text{day2} + \cdots + \gamma_6 \text{day7} + \epsilon \quad (6)$$

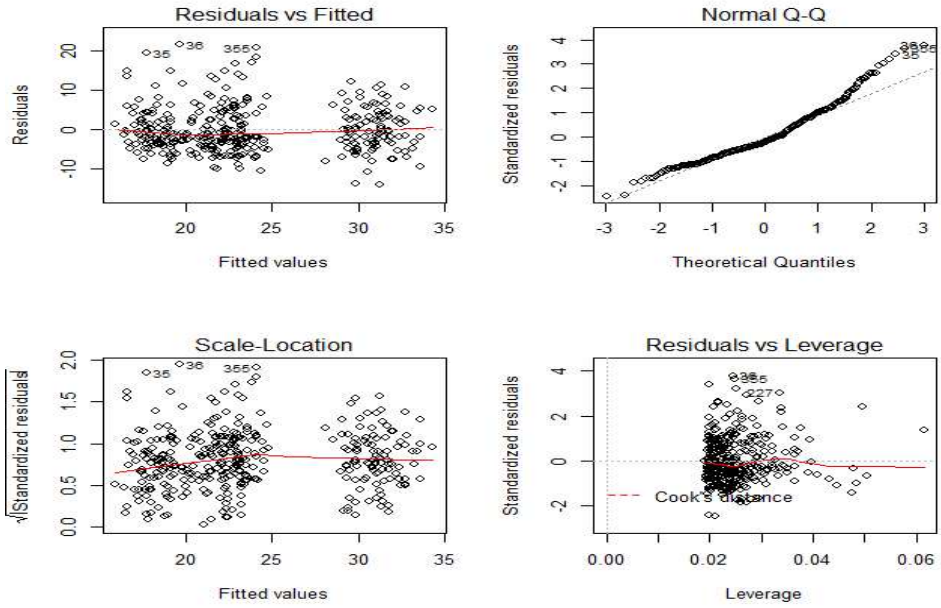


그림 4. 최종모형의 가정 확인을 위한 그래프들

최종모형은 식 (6)으로 표현되며, <그림 4>의 그래프와 (x, r) 그래프를 살펴보았을 때 분산의 동질성 가정과 선형성을 어느 정도 만족하는 것처럼 보이며, leverage vs. DFITS 그래프를 보았을 때에도 눈에 띄는 영향점은 없었다. 다중공선성을 확인하기 위해 분산팽창지수(VIF)를 구해본 결과 <표 5>와 같이 작은 값들로 나타나 다중공선성은 존재하지 않음을 알 수 있었다.

표 5. VIF

	GVIF	Df
최대순간풍속	1.02	1
1시간 최대일사량	1.02	1
요일변수	1.02	6

마지막으로 자기상관 유무를 확인하기 위해 $H_0 : \rho = 0$ vs $H_1 : \rho > 0$을 가설로 한 검정의 결과는 다음과 같았다.

- 더빈-왓슨 검정 : *d-value* = 0.50, p < .001
- runs 검정 : *z-value* = -12.88, p < 0.001

두 검정 모두 귀무가설을 기각할 수 있으므로 유의미한 양의 자기상관이 존재함을 알 수 있으며 lag 1 표본자기상관계수가 .73으로 매우 강한 상관관계가 있다. 요일변수를 추가했기 때문에 lag 7 표본자기상관계수는 .19로 비교적 낮게 나타났지만 여전히 인접한 오차항들 사이의 자기상관문제가 강하게 나타나고 있는 것이다.

최종적인 모형의 분석 결과는 <표 6>과 같다. 요일변수는 매우 유의하게 나타났고, 토요일의 관객 수가 평균적으로 가장 많으며 월요일과 화요일의 관객 수가 평균적으로 가장 적다.

이처럼 요일 변수가 매우 유의하긴 하지만 본 연구에서는 연구의 초기 목적이었던 날씨 변수에 집중하고자 한다. 순간최대풍속은 5% 유의수준 하에서 유의하게 나타났으며 다른 변수들이 변하지 않을 때, 순간최대풍속이 1(m/s) 증가하면 $\sqrt{관객수}$는 0.69 감소한다. 1시간 최대일사량은 10% 유의수준 하에서 유의하게 나타났으며 다른 변수들이 변하지 않을 때, 1시간 최대일사량이 1(MJ/m2) 증가하면 $\sqrt{관객수}$는 0.71 감소한다.

표 6. 날씨와 요일에 따른 영화 관객수 다중회귀분석

설명 변수	$\hat{\beta}$	*SE*	*t*	*p*	95% CI	
					Lower	*Upper*
상수	22.69	1.71	13.23	< .001	19.32	26.06
순간최대풍속	**−0.69**	**0.31**	**−2.21**	**.03**	**−1.30**	**−0.07**
1시간 최대일사량	**−0.71**	**0.39**	**−1.82**	**.07**	**−1.47**	**0.06**
day2 (화요일)	0.43	1.13	−0.38	.70	−1.79	2.65
day3 (수요일)	4.64	1.14	4.08	< .001	2.40	6.87
day4 (목요일)	3.59	1.14	3.16	.002	1.35	5.82
day5 (금요일)	4.80	1.14	4.21	< .001	2.56	7.04
day6 (토요일)	13.45	1.14	11.77	< .001	11.21	15.70
day7 (일요일)	11.83	1.13	10.45	< .001	9.60	14.05
$F(8,352) = 34$, p < .001						
R^2 = .44, Adj R^2 = .42						

4. 결론 및 한계

지금까지 날씨에 따라 관객 수가 유의미한 차이를 보이는지에 대해 다중회귀분석을 통하여 분석해보았다. 주로 변수를 제거하거나 추가해나가면서 회귀분석의 가정을 만족시키는 것에 초점을 맞추어 분석을 진행하였고, 그 결과 6개의 기상 데이터 중 최대 순간풍속과 1시간 최대 일사량이 유의미한 변수로 최종 선택되었다. 요일 더미 변수들이 존재하는 상황에서 이들이 증가하면 관객 수는 줄어드는 관계가 있으며, 이는 바람이나 햇빛이 강할수록 야외활동 중에서도 실내에서 오랜 시간을 보낼 수 있는 특성을 가진 극장 영화관람을 선호하는 사람들의 경향성이 반영되었다고 보는 것이 타당할 것이다.

결과적으로 본 연구를 시작하기 전 기대했던 바와 달리 기온이나 습도에 대한 변수는 유의미하게 나타나지 않았다. 기온이 극단적으로 높거나 낮은 경우, 또는 비가 오는 경우 실내의 쾌적한 환경에서 시간을 보낼 수 있는 극장 영화관람을 선호할 것이라는 예상을 하였으나 여름과 겨울의 데이터를 한꺼번에 고려하다 보니 기온의 영향을 유의미하게 밝히지 못한 것으로 생각된다. 특히, 기온은 여름만을 고려한 비슷한 주제의 선행 연구에서는 유의미한 관계를 보인 바가 있기 때문에 향후 여름과 겨울의 데이터를 분리하여 분석하거나 다른 방법을 활용하여 기온과 관객 수와의 관계를 다시 한번 분석해본다면 좋을 것이다.

본 연구에는 여러 가지 한계점들이 존재한다. 먼저, 최종 모형의 수정결정계수가 .42에 불과하여 모형의 설명력이 비교적 낮다. 그리고 독립 변수와 종속 변수의 범위가 달리 설정되어 서울의 날씨와 전국의 총관객 수 데이터를 사용하였기 때문에 연구결과에 대한 신뢰성이 낮을 수 있다. 마지막으로 자기상관 문제를 결국 해결하지 못하였다. lag 1과 lag 7의 자기상관계수가 동시에 높게 나옴으로 인해 자기상관을 줄이긴 하였으나 모두 제거하지는 못하였고, 최종 모형에서도 상당히 높은 자기상관이 존재함을 확인할 수 있었다.

따라서 향후 비슷한 주제로 연구를 진행한다면 앞서 말한 한계점들을 해결하는 것에 초점을 맞추어 전개해나갈 수 있을 것이며, 이들을 보완한다면 독립 변수와 종속 변수 사이의 관계를 더욱 명확하게 이끌어낼 수 있으리라 기대된다.

참고문헌

영화진흥위원회. (2020). 2019년 한국 영화산업 결산 보고서. URL: https://www.kofic.or.kr/kofic/business/board/selectBoardDetail.do?boardNumber=2&boardSeqNumber=49961#none

김형호. (2016. 08. 01). [김형호의 영화시장] 찜통더위 길수록 영화시장 커진다. <매일경제>. URL: https://www.mk.co.kr/star/hot-issues/view/2016/08/547174

이문원. (2020. 06. 02). [문화칼럼] 1인당 영화관람 횟수 세계 1위 한국은 영화광들의 나라다?. <자유기업원>. URL: https://www.cfe.org/20200602_22791

Appendix

```
# Processing the Data ----------------------------------------------------

## load data
raw_data = read.csv("data.csv", header = T)

## change the column names
names(raw_data) <- c('date', 'day', 'aud', 'avgtem', 'lowtem', 'hitem', 'rain',
                     'avgWS', 'maxWS', 'avgHum', 'sumSun','sumSR', 'maxSun',
                     'tem15', 'maxSn', 'avgCl', 'dust')
data = raw_data
head(data)
```

```
##         date day     aud avgtem lowtem hitem rain avgWS maxWS avgHum sumSun
## 1 2019-01-01   2 1163267   -5.0   -8.2  -0.6   NA   2.1   4.3   49.5    7.5
## 2 2019-01-02   3  353349   -4.9   -8.8   0.2   NA   1.7   3.6   42.8    8.7
## 3 2019-01-03   4  394614   -3.5   -8.4   3.2   NA   1.4   2.9   38.8    8.7
## 4 2019-01-04   5  388232   -1.1   -6.2   4.1   NA   1.2   3.0   55.5    3.9
## 5 2019-01-05   6  820944   -2.8   -5.5   1.1   NA   2.2   4.3   40.3    8.6
## 6 2019-01-06   7  725846   -2.8   -6.3   2.7   NA   1.2   3.2   35.0    7.7
##   sumSR maxSun tem15 maxSn avgCl dust
## 1  7.84   1.42   9.1    NA   3.4   41
## 2 10.48   1.81   8.9    NA   0.0   38
## 3 10.28   1.79   8.8    NA   0.1   41
## 4  6.20   1.27   8.7    NA   5.5   77
## 5 10.05   1.78   8.6    NA   0.5   73
## 6  9.64   1.79   8.4    NA   3.1   47
```

```
## in thausands
data['aud'] = data['aud']/1000

## check NA values
## rain 226, maxSn 359, dust 27, sumSun 1, sumSR 2, maxSun 1, tem15 1
apply(is.na(data), 2, sum)
```

```
##   date    day    aud avgtem lowtem  hitem   rain  avgWS  maxWS avgHum sumSun
##      0      0      0      0      0      0    226      0      0      0      1
##  sumSR maxSun  tem15  maxSn  avgCl   dust
##      2      1      1    359      0     27
```

```
## delete the data indexed 240, 241, 284 which has NA value
data[c(240, 241, 284),]
```

```
##           date day     aud avgtem lowtem hitem rain avgWS maxWS avgHum sumSun
## 240 2019-08-28   3 688.516   26.1   23.6  30.2   NA   1.9   4.3   66.2     NA
```

```
## 241 2019-08-29   4 336.735   23.4   20.1  26.4 36.9   2.2   7.5   77.1    4.9
## 284 2019-10-11   5 393.823   18.8   13.0  26.1   NA   1.9   4.9   60.0   10.3
##     sumSR maxSun tem15 maxSn avgCl dust
## 240    NA     NA  24.1    NA   5.9   NA
## 241    NA   1.94  24.1    NA   5.6   NA
## 284 16.44   2.45    NA    NA   0.9   30
```

```
data = data[-c(240, 241, 284),]
rownames(data) <- NULL  # initialize index

#### drop variables with many NA values
## maxSn, dust: drop
## rain: replace

#install.packages("PerformanceAnalytics")
library("PerformanceAnalytics")
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

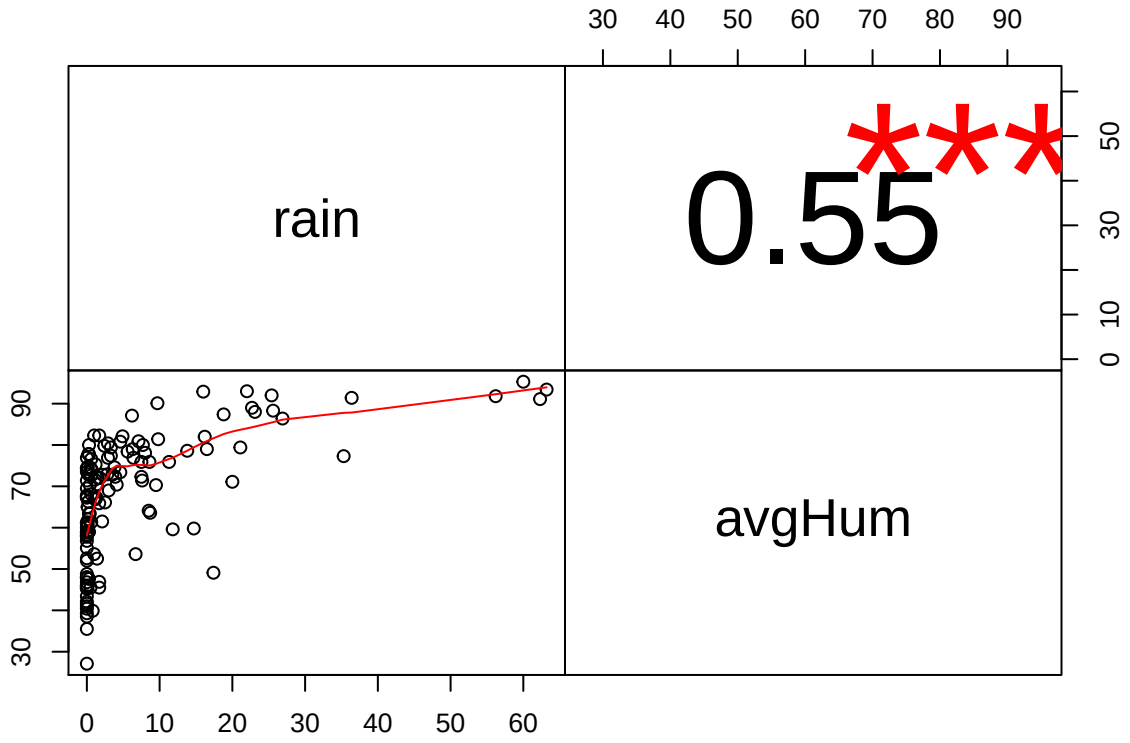
```
##
## Attaching package: 'PerformanceAnalytics'
```

```
## The following object is masked from 'package:graphics':
##
##     legend
```

```
attach(data)

## correlation between rain & average humidity (r = .55, p < .001)
cor.test(data[!is.na(rain), "rain"], data[!is.na(rain), "avgHum"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  data[!is.na(rain), "rain"] and data[!is.na(rain), "avgHum"]
## t = 7.6403, df = 136, p-value = 3.481e-12
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4193090 0.6551262
## sample estimates:
##       cor
## 0.5480152
```

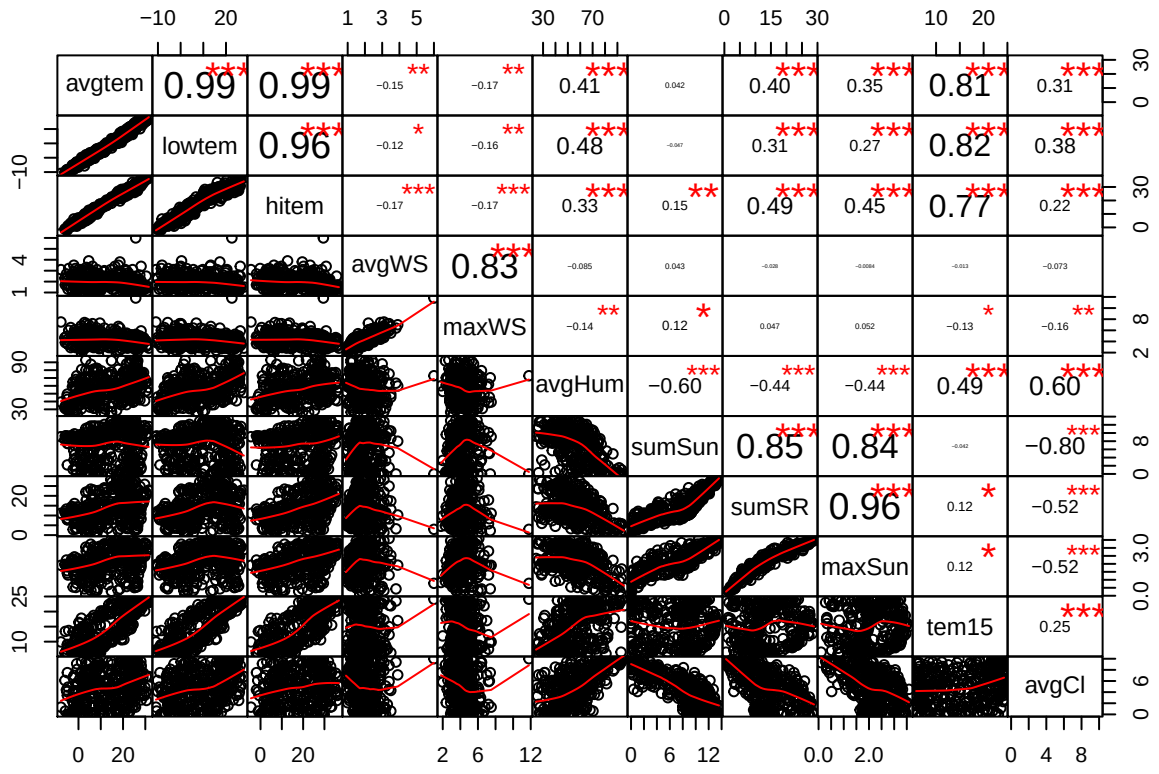
```
chart.Correlation(data[!is.na(rain), c("rain", 'avgHum')], histogram = FALSE)
```




```
## drop variables with many NA values
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:xts':
##
##     first, last

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
data = data %>% select(-c(rain, maxSn, dust))

## correlation between all independent variables
chart.Correlation(data[,4:14], histogram=FALSE)
```




```
## through simple linear regression, choose one variable
summary(lm(sqrt(aud)~avgtem, data=data)) #0.308
```

```
##
## Call:
## lm(formula = sqrt(aud) ~ avgtem, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.859  -6.179  -1.653   5.503  21.532
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 23.22850    0.66268   35.05   <2e-16 ***
## avgtem       0.03996    0.03916    1.02    0.308
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.58 on 360 degrees of freedom
## Multiple R-squared:  0.002884,   Adjusted R-squared:  0.0001138
## F-statistic: 1.041 on 1 and 360 DF,  p-value: 0.3083
```

```
summary(lm(sqrt(aud)~lowtem, data=data)) #0.198
```

```
##
```

```
## Call:
## lm(formula = sqrt(aud) ~ lowtem, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.700  -6.233  -1.566   5.541  21.664
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 23.31051    0.53351   43.69   <2e-16 ***
## lowtem       0.04959    0.03844    1.29    0.198
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.574 on 360 degrees of freedom
## Multiple R-squared:  0.004603,   Adjusted R-squared:  0.001838
## F-statistic: 1.665 on 1 and 360 DF,  p-value: 0.1978
```

```
summary(lm(sqrt(aud)~hitem, data=data)) #0.487
```

```
##
## Call:
## lm(formula = sqrt(aud) ~ hitem, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.887  -6.185  -1.788   5.333  21.392
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 23.27155    0.81877  28.423   <2e-16 ***
## hitem        0.02684    0.03860   0.695    0.487
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.586 on 360 degrees of freedom
## Multiple R-squared:  0.001341,   Adjusted R-squared:  -0.001433
## F-statistic: 0.4835 on 1 and 360 DF,  p-value: 0.4873
```

```
summary(lm(sqrt(aud)~avgWS, data=data)) #0.0319*
```

```
##
## Call:
## lm(formula = sqrt(aud) ~ avgWS, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.420  -5.800  -1.815   5.465  19.707
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  26.5644     1.3571  19.575   <2e-16 ***
## avgWS        -1.4308     0.6643  -2.154   0.0319 *
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.543 on 360 degrees of freedom
## Multiple R-squared:  0.01272,    Adjusted R-squared:  0.009981
## F-statistic: 4.639 on 1 and 360 DF,  p-value: 0.03191
```

```
summary(lm(sqrt(aud)~maxWS, data=data)) #0.00495**
```

```
##
## Call:
## lm(formula = sqrt(aud) ~ maxWS, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.720  -5.944  -1.637   5.224  19.940
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  28.2659     1.6384  17.252  < 2e-16 ***
## maxWS        -1.0516     0.3719  -2.828  0.00495 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.508 on 360 degrees of freedom
## Multiple R-squared:  0.02173,    Adjusted R-squared:  0.01901
## F-statistic: 7.997 on 1 and 360 DF,  p-value: 0.004947
```

```
summary(lm(sqrt(aud)~sumSun, data=data)) #0.18
```

```
##
## Call:
## lm(formula = sqrt(aud) ~ sumSun, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.592  -6.213  -1.689   5.612  20.579
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 24.70312    0.80101  30.840   <2e-16 ***
## sumSun      -0.13384    0.09958  -1.344     0.18
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.572 on 360 degrees of freedom
## Multiple R-squared:  0.004993,   Adjusted R-squared:  0.002229
## F-statistic: 1.807 on 1 and 360 DF,  p-value: 0.1798
```

```
summary(lm(sqrt(aud)~sumSR, data=data)) #0.22
```

```
##
```

```
## Call:
## lm(formula = sqrt(aud) ~ sumSR, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.268  -6.159  -1.650   5.519  20.562
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 24.74918    0.89210  27.743   <2e-16 ***
## sumSR       -0.07095    0.05778  -1.228     0.22
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.575 on 360 degrees of freedom
## Multiple R-squared:  0.004172,   Adjusted R-squared:  0.001405
## F-statistic: 1.508 on 1 and 360 DF,  p-value: 0.2202
```

```
summary(lm(sqrt(aud)~maxSun, data=data)) #0.104
```

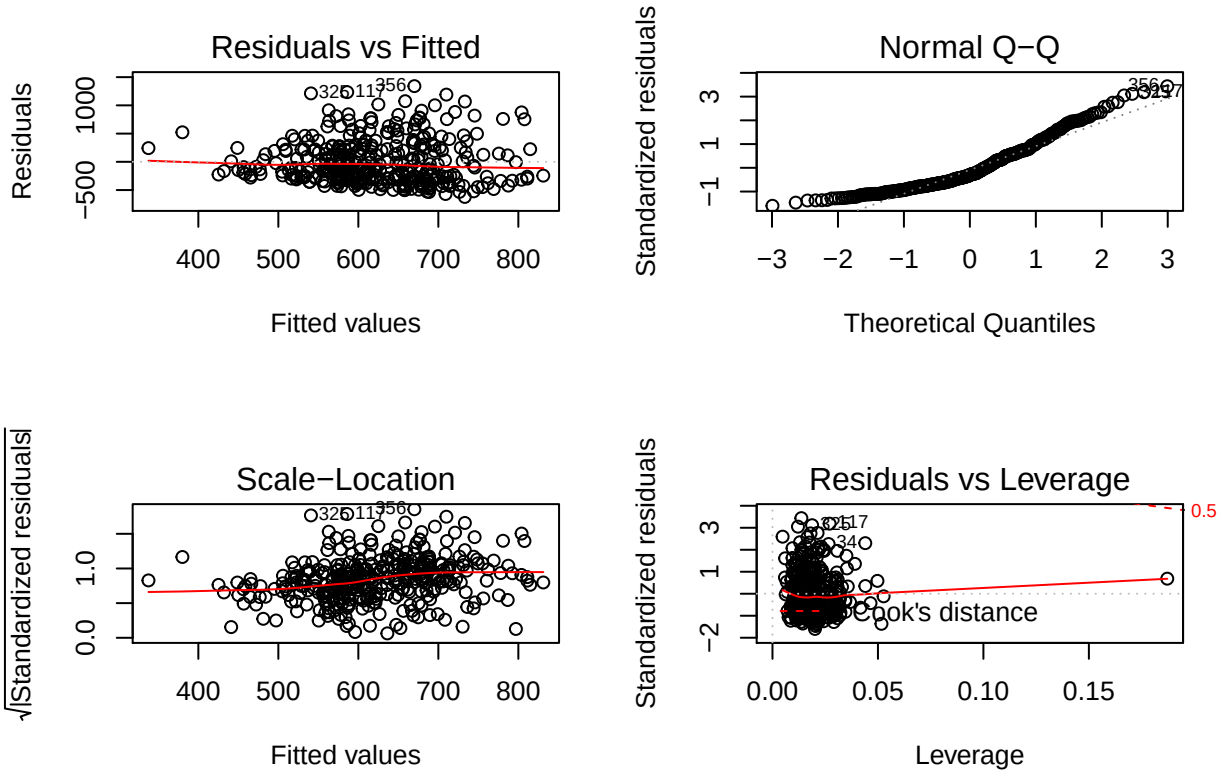
```
##
## Call:
## lm(formula = sqrt(aud) ~ maxSun, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.190  -6.194  -1.754   5.387  20.494
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  25.5122     1.1421  22.337   <2e-16 ***
## maxSun       -0.8208     0.5041  -1.628    0.104
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.563 on 360 degrees of freedom
## Multiple R-squared:  0.00731,    Adjusted R-squared:  0.004553
## F-statistic: 2.651 on 1 and 360 DF,  p-value: 0.1044
```

```
# Model 1 (full model) ------------------------------------------------------

m1 = lm(aud ~ lowtem + maxWS +  avgHum +
                maxSun + tem15 + avgCl, data=data)
summary(m1)
```

```
##
## Call:
## lm(formula = aud ~ lowtem + maxWS + avgHum + maxSun + tem15 +
##     avgCl, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -621.8 -296.9 -117.2  228.2 1340.6
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1069.844    223.325   4.791 2.45e-06 ***
## lowtem         6.194      4.906   1.262  0.20763
## maxWS        -52.744     19.841  -2.658  0.00821 **
## avgHum        -1.681      2.113  -0.796  0.42678
## maxSun       -59.183     46.498  -1.273  0.20391
## tem15         -5.474      6.243  -0.877  0.38120
## avgCl          5.084     11.232   0.453  0.65110
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 392.7 on 355 degrees of freedom
## Multiple R-squared:  0.04046,    Adjusted R-squared:  0.02424
## F-statistic: 2.495 on 6 and 355 DF,  p-value: 0.02234
```

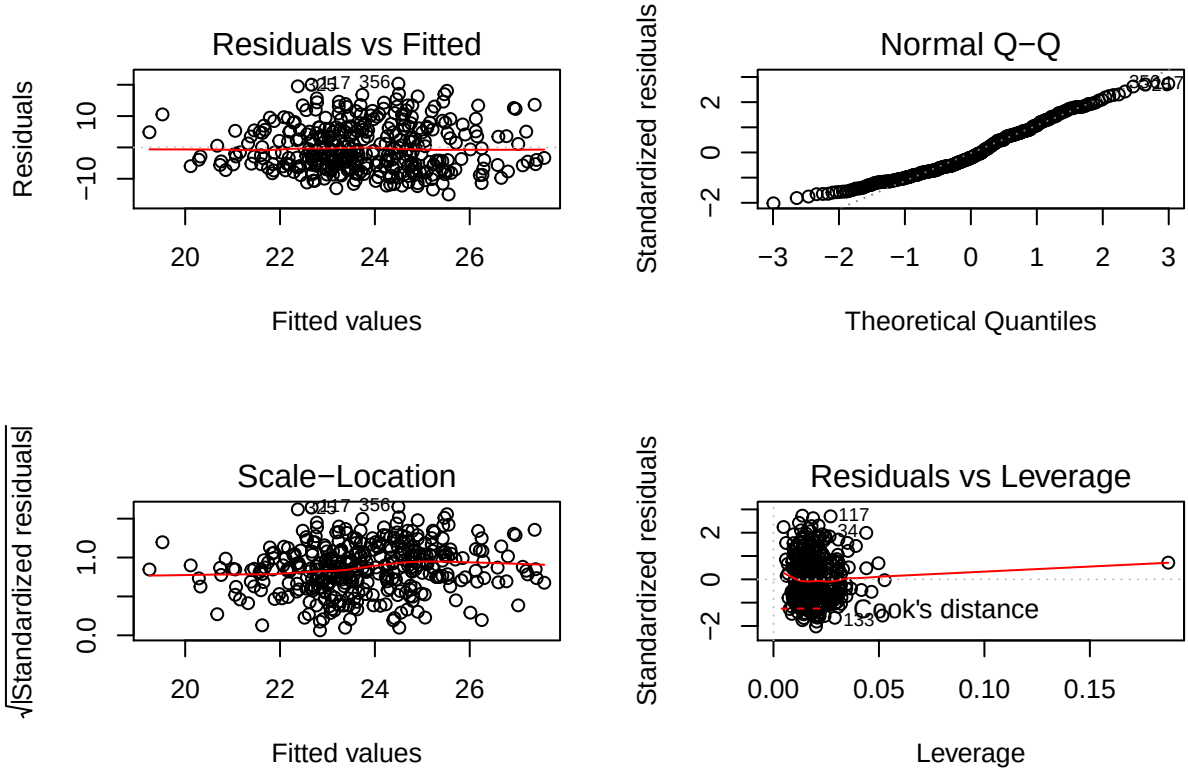
```
## checking assumptions
par(mfrow = c(2, 2))
plot(m1) # homogeneous variance asssumption is violated.
```



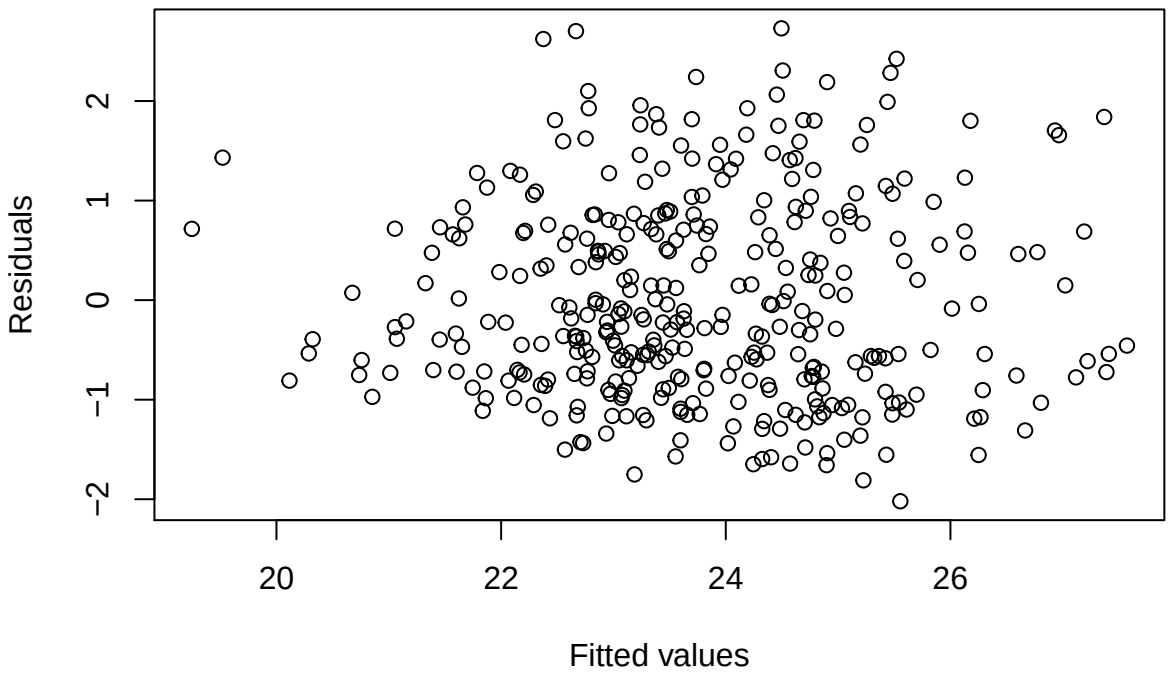

```
# Model 2 (sqrt) ------------------------------------------------------------

m2 = lm(sqrt(aud) ~ lowtem + maxWS + avgHum +
                maxSun + tem15 + avgCl, data=data)
summary(m2)
```

```
##
## Call:
## lm(formula = sqrt(aud) ~ lowtem + maxWS + avgHum + maxSun + tem15 +
##     avgCl, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.005  -5.732  -1.594   5.312  20.345
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 32.46580    4.26646   7.610  2.5e-13 ***
## lowtem       0.12308    0.09373   1.313   0.1900
## maxWS       -0.91804    0.37905  -2.422   0.0159 *
## avgHum      -0.03553    0.04036  -0.880   0.3793
## maxSun      -1.35761    0.88830  -1.528   0.1273
## tem15       -0.07853    0.11927  -0.658   0.5107
## avgCl        0.03739    0.21458   0.174   0.8618
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.503 on 355 degrees of freedom
## Multiple R-squared:  0.03666,    Adjusted R-squared:  0.02038
## F-statistic: 2.252 on 6 and 355 DF,  p-value: 0.03805
```

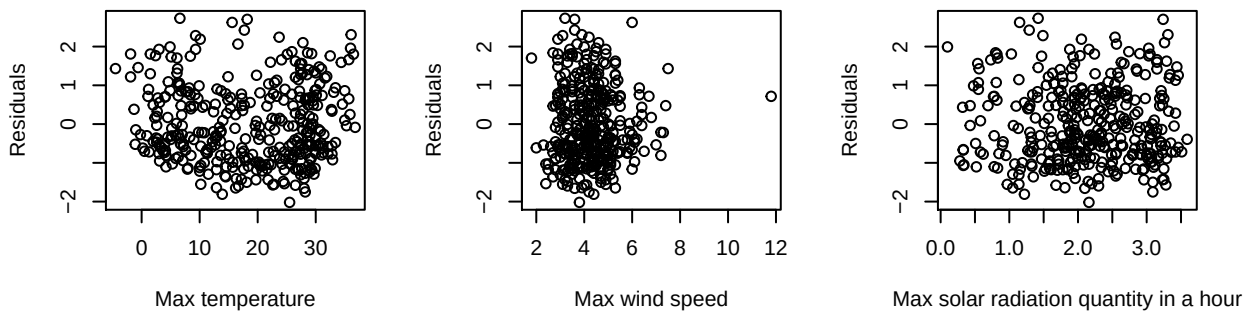
```
X = model.matrix(m2)
n = nrow(X)
p = ncol(X)

## checking assumptions
par(mfrow = c(2, 2))
plot(m2)
```

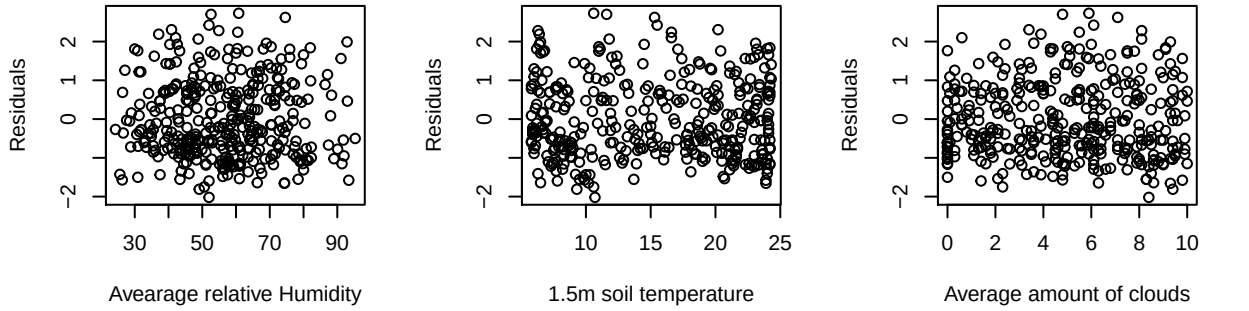



```
## (y_hat, r)
par(mfrow = c(1, 1))
res2 = rstandard(m2)
plot(m2$fitted.values, res2,
     xlab = 'Fitted values', ylab='Residuals')
```

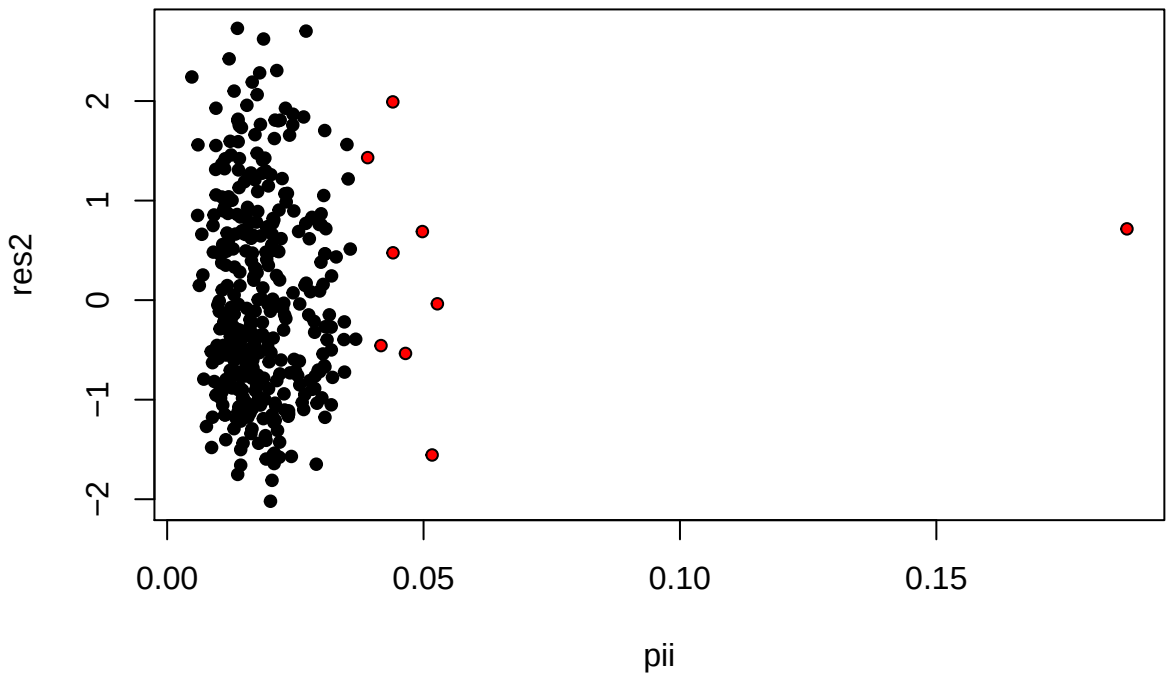



```
## (x, r) well scattered
par(mfrow = c(2, 3))
plot(hitem, res2, xlab="Max temperature", ylab="Residuals")
plot(maxWS, res2, xlab="Max wind speed", ylab="Residuals")
plot(maxSun, res2, xlab="Max solar radiation quantity in a hour", ylab="Residuals")
plot(avgHum, res2, xlab="Avearage relative Humidity", ylab="Residuals")
plot(tem15, res2, xlab="1.5m soil temperature", ylab="Residuals")
plot(avgCl, res2, xlab="Average amount of clouds", ylab="Residuals")
```

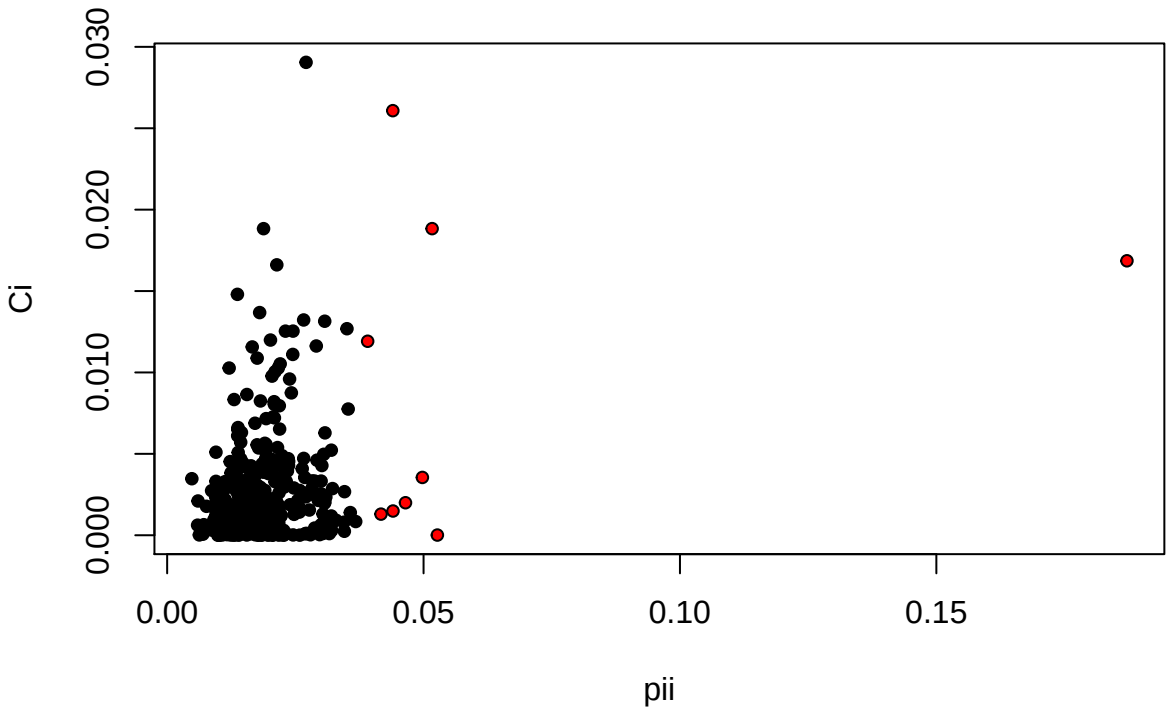





```
## checking outlier
par(mfrow = c(1, 1))

# 1.leverage point
pii = influence(m2)$hat
lev.idx = pii > 2*p/n

color = rep("black", len=n)
color[lev.idx] = 'red'

plot(pii, res2, type='n')
points(pii, res2, pch=21, cex=0.8, bg=color)
```



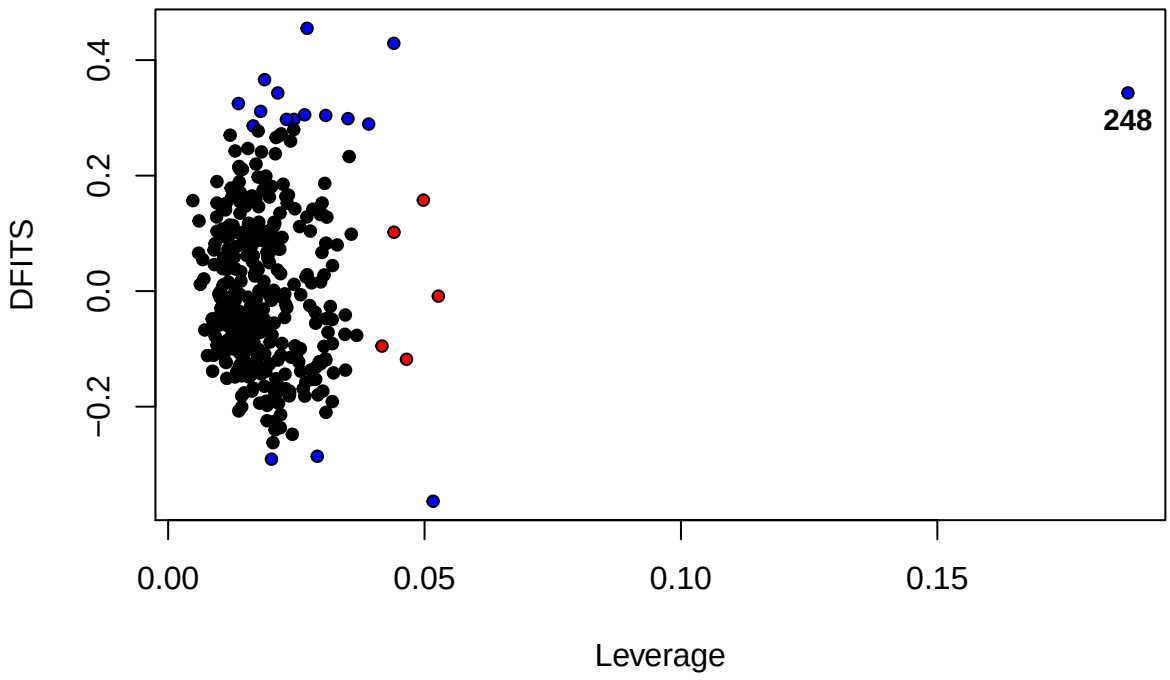

```
# 2.cooks distance
Ci = cooks.distance(m2)
plot(pii, Ci, type='n')
points(pii, Ci, pch=21, cex=0.8, bg=color)
```




```
sum(Ci > qf(0.5,p,n-p)) # none
```

```
## [1] 0
```

```
# 3.dfits
dfits = dffits(m2)
inf.idx = abs(dfits) >= 2*sqrt(p/(n-p))
color[inf.idx] = 'blue'


# index 248: influential point far from others
plot(pii, dfits, type='n',
     xlab = 'Leverage', ylab = 'DFITS')
points(pii, dfits, pch=21, cex=0.8, bg=color)
text(dfits[which.max(pii)]~pii[which.max(pii)],
     labels=names(lev.idx[which.max(pii)]),
     cex=0.9, font=2, pos=1)
```

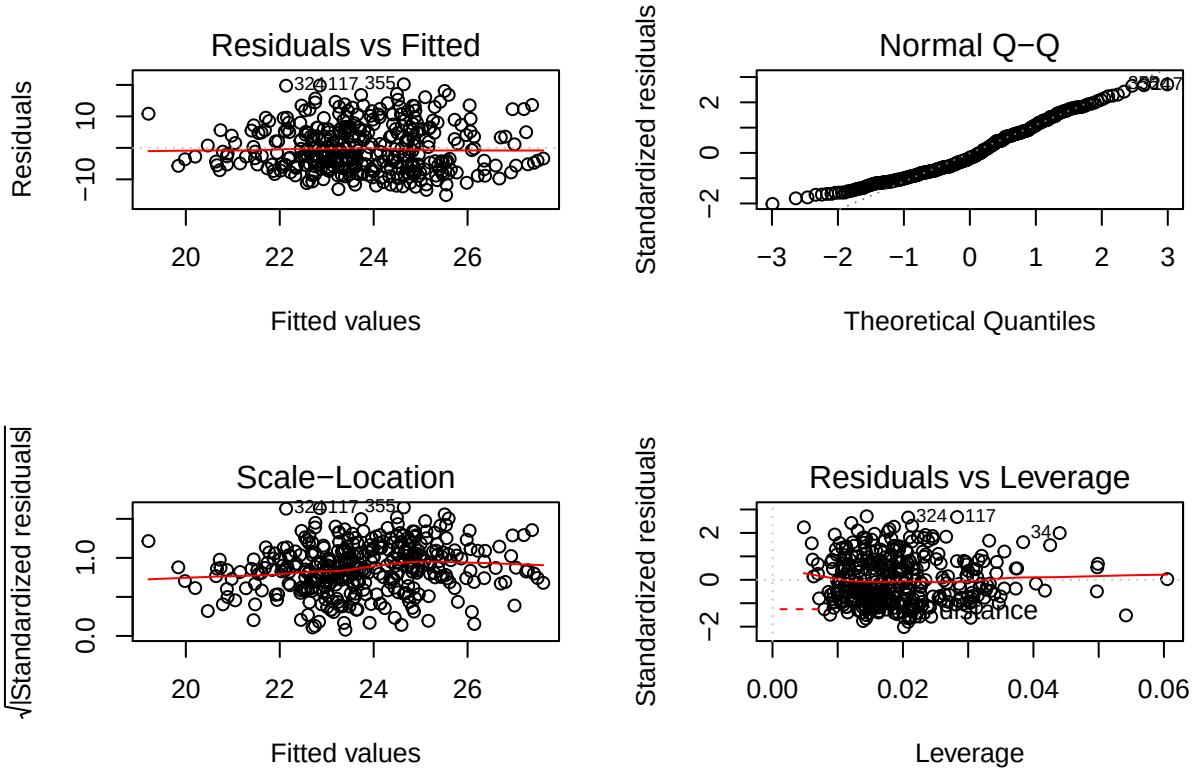



```
# Model 3 (eliminate the influential point) -------------------------------

data.rem = data[-248,] # eliminate the influential point
rownames(data.rem) <- NULL

m3 = lm(sqrt(aud) ~ lowtem + maxWS + maxSun +
          avgHum + tem15 + avgCl, data=data.rem)
summary(m3)
```

```
##
## Call:
## lm(formula = sqrt(aud) ~ lowtem + maxWS + maxSun + avgHum + tem15 +
##     avgCl, data = data.rem)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.993  -5.707  -1.419   5.494  20.199
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 32.74916    4.28773   7.638 2.08e-13 ***
## lowtem       0.11647    0.09425   1.236   0.2174
## maxWS       -1.03917    0.41539  -2.502   0.0128 *
## maxSun      -1.27419    0.89653  -1.421   0.1561
## avgHum      -0.03318    0.04052  -0.819   0.4135
```

```
## tem15       -0.07966    0.11936  -0.667   0.5050
## avgCl        0.03448    0.21477   0.161   0.8725
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.508 on 354 degrees of freedom
## Multiple R-squared:  0.03804,    Adjusted R-squared:  0.02174
## F-statistic: 2.333 on 6 and 354 DF,  p-value: 0.03186
```

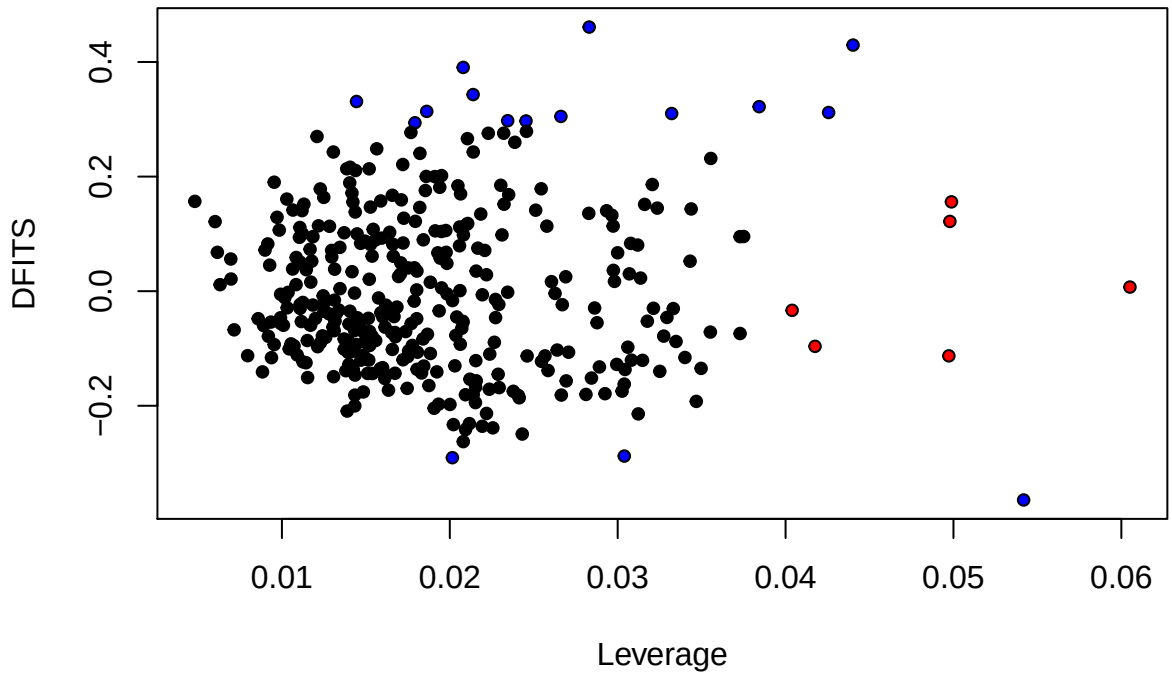
```r
## checking assumptions
par(mfrow = c(2, 2))
plot(m3) # Ok
```




```r
## checking outlier
par(mfrow = c(1, 1))

# 1.leverage point
pii = influence(m3)$hat
lev.idx = pii > 2*p/n
color = rep("black", len=n)
color[lev.idx] = 'red'

# 2.cooks distance
Ci = cooks.distance(m3)
sum(Ci > qf(0.5,p,n-p)) # none
```

```
## [1] 0
```

```
# 3.dfits
dfits = dffits(m3)
inf.idx = abs(dfits) >= 2*sqrt(p/(n-p))
color[inf.idx] = 'blue'

# there is no influential point which is highly different from others
plot(pii, dfits, type='n',
     xlab = 'Leverage', ylab = 'DFITS')
points(pii, dfits, pch=21, cex=0.8, bg=color)
```



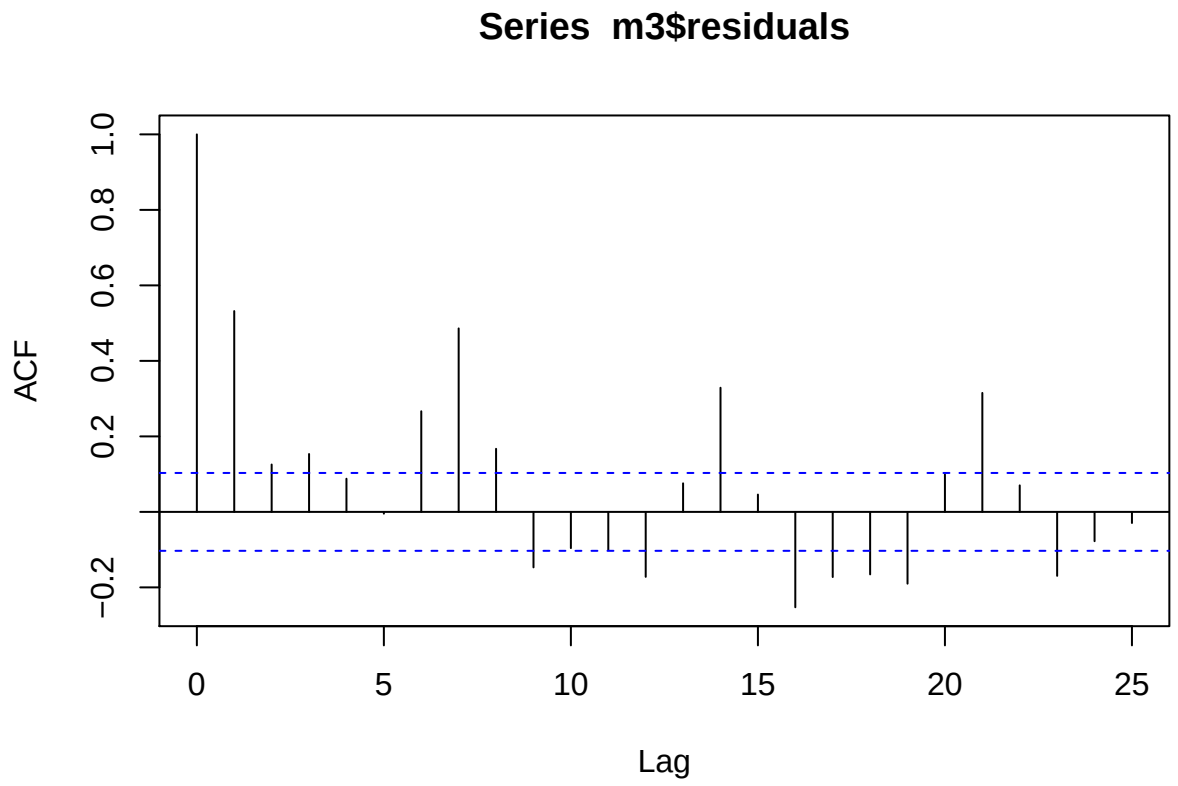

```
## checking autocorrelation
# Durbin-Watson test
library(lmtest)
dwtest(m3, alternative = 'greater')
```

```
##
##  Durbin-Watson test
##
## data:  m3
## DW = 0.92423, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
# runs test
library(randtests)
runs.test(m3$residuals, alternative="left.sided", plot = FALSE)
```

```
##
##  Runs Test
##
## data:  m3$residuals
## statistic = -7.6001, runs = 109, n1 = 180, n2 = 180, n = 360, p-value =
## 1.48e-14
## alternative hypothesis: trend
```

```r
# Model 4 (AR(1)) ------------------------------------------------------------

## sample autocorrelation coefficient
acf(m3$residuals, plot=TRUE)
```




```r
acf(m3$residuals, plot=FALSE)
```

```
##
## Autocorrelations of series 'm3$residuals', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000  0.532  0.126  0.153  0.088 -0.005  0.267  0.486  0.167 -0.147 -0.096
```

```
##     11     12     13     14     15     16     17     18     19     20     21
## -0.101 -0.172  0.076  0.329  0.046 -0.253 -0.173 -0.166 -0.190  0.099  0.315
##     22     23     24     25
##  0.070 -0.169 -0.078 -0.029
```

```
rho <- acf(m3$residuals, plot=FALSE)[1]$acf[1]
n <- nrow(data.rem)

# Fit a linear model with transformed variables
taud <- sqrt(data.rem$aud[2:n]) - rho*sqrt(data.rem$aud[1:(n-1)])
thitem <- data.rem$hitem[2:n] - rho*data.rem$hitem[1:(n-1)]
tmaxWS <- data.rem$maxWS[2:n] - rho*data.rem$maxWS[1:(n-1)]
tmaxSun <- data.rem$maxSun[2:n] - rho*data.rem$maxSun[1:(n-1)]
tavgHum <- data.rem$avgHum[2:n] - rho*data.rem$avgHum[1:(n-1)]
ttem15 <- data.rem$tem15[2:n] - rho*data.rem$tem15[1:(n-1)]
tavgCl <- data.rem$avgCl[2:n] - rho*data.rem$avgCl[1:(n-1)]
m4 <- lm(taud ~ thitem + tmaxWS + tmaxSun +
           tavgHum + ttem15 + tavgCl)
summary(m4)
```

```
##
## Call:
## lm(formula = taud ~ thitem + tmaxWS + tmaxSun + tavgHum + ttem15 +
##     tavgCl)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.8525  -3.7111  -0.5882   4.3973  21.3609
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 12.16942    1.64612   7.393 1.05e-12 ***
## thitem       0.05607    0.10628   0.528    0.598
## tmaxWS      -0.50848    0.34523  -1.473    0.142
## tmaxSun     -0.25922    0.79719  -0.325    0.745
## tavgHum     -0.02146    0.03602  -0.596    0.552
## ttem15      -0.02442    0.16792  -0.145    0.884
## tavgCl       0.20296    0.16601   1.223    0.222
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.301 on 353 degrees of freedom
## Multiple R-squared:  0.02023,    Adjusted R-squared:  0.003577
## F-statistic: 1.215 on 6 and 353 DF,  p-value: 0.2979
```

```
acf(m4$residuals, plot=FALSE)
```

```
##
## Autocorrelations of series 'm4$residuals', by lag
##
##     0     1     2     3     4     5     6     7     8     9    10
## 1.000 0.148 -0.280 0.112 0.053 -0.279 0.134 0.581 0.054 -0.322 0.024
##    11    12    13    14    15    16    17    18    19    20    21
```

```
##  0.010 -0.294  0.043  0.512  0.037 -0.365 -0.001 -0.010 -0.302  0.090  0.468
##     22     23     24     25
##  0.022 -0.312  0.016  0.051
```

```
dwtest(m4, alternative = 'greater')
```

```
##
##  Durbin-Watson test
##
## data:  m4
## DW = 1.693, p-value = 0.001568
## alternative hypothesis: true autocorrelation is greater than 0
```

```
runs.test(m4$residuals, alternative="left.sided", plot = FALSE)
```

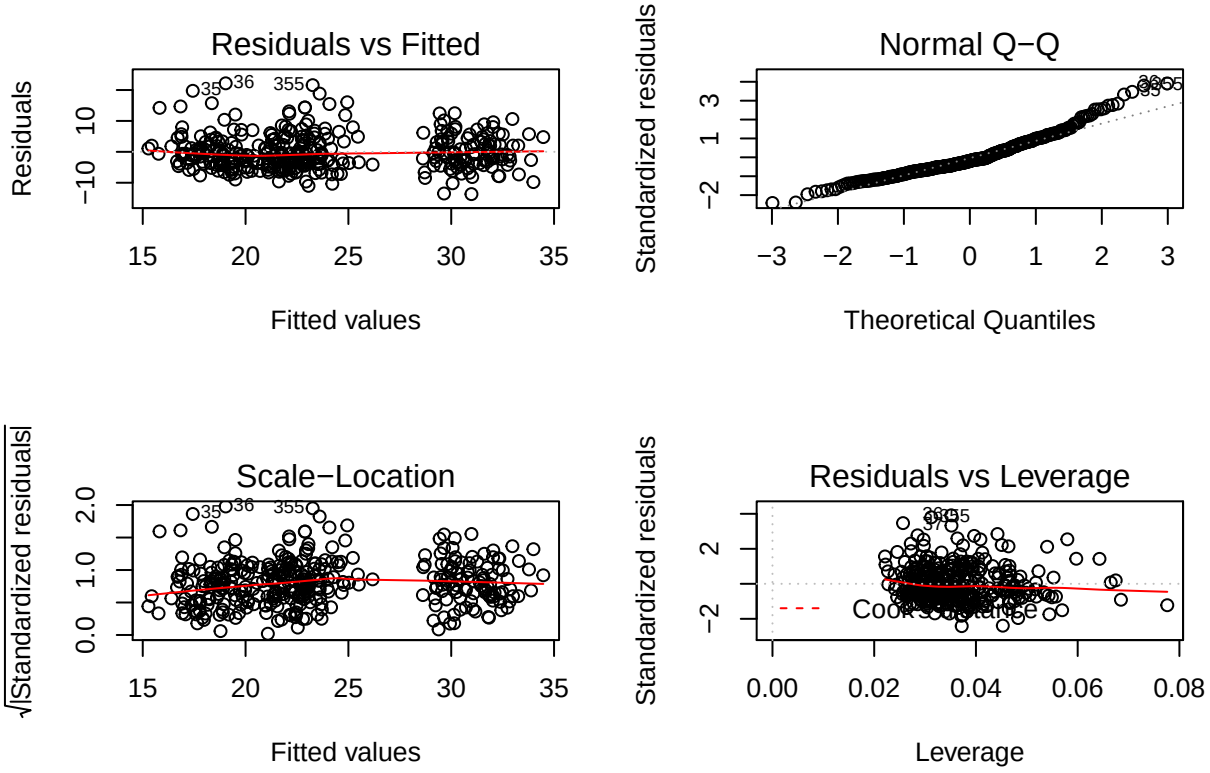
```
##
##  Runs Test
##
## data:  m4$residuals
## statistic = -2.85, runs = 154, n1 = 180, n2 = 180, n = 360, p-value =
## 0.002186
## alternative hypothesis: trend
```

```
# Model 5 (add the variable) ------------------------------------------------

m5 = lm(sqrt(aud) ~ lowtem + maxWS + maxSun +
          avgHum + tem15 + avgCl + as.factor(day), data=data.rem)

summary(m5)
```

```
##
## Call:
## lm(formula = sqrt(aud) ~ lowtem + maxWS + maxSun + avgHum + tem15 +
##     avgCl + as.factor(day), data = data.rem)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.675  -3.725  -1.097   3.273  22.166
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     25.051051   3.367565   7.439 8.00e-13 ***
## lowtem           0.123453   0.072599   1.700  0.08994 .
## maxWS           -0.573406   0.321595  -1.783  0.07546 .
## maxSun          -1.476372   0.691700  -2.134  0.03351 *
## avgHum          -0.007625   0.031252  -0.244  0.80738
## tem15           -0.080992   0.091811  -0.882  0.37830
## avgCl           -0.160655   0.165930  -0.968  0.33361
## as.factor(day)2  0.544426   1.129259   0.482  0.63003
## as.factor(day)3  4.746981   1.141203   4.160 4.02e-05 ***
## as.factor(day)4  3.720155   1.140747   3.261  0.00122 **
## as.factor(day)5  4.953099   1.143465   4.332 1.94e-05 ***
```

```
## as.factor(day)6 13.573764   1.147844  11.825  < 2e-16 ***
## as.factor(day)7 11.906343   1.136221  10.479  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.768 on 348 degrees of freedom
## Multiple R-squared:  0.442,  Adjusted R-squared:  0.4227
## F-statistic: 22.97 on 12 and 348 DF,  p-value: < 2.2e-16
```

```
X = model.matrix(m5)
n = nrow(X)
p = ncol(X)

## checking assumptions
par(mfrow = c(2, 2))
plot(m5) # Ok
```

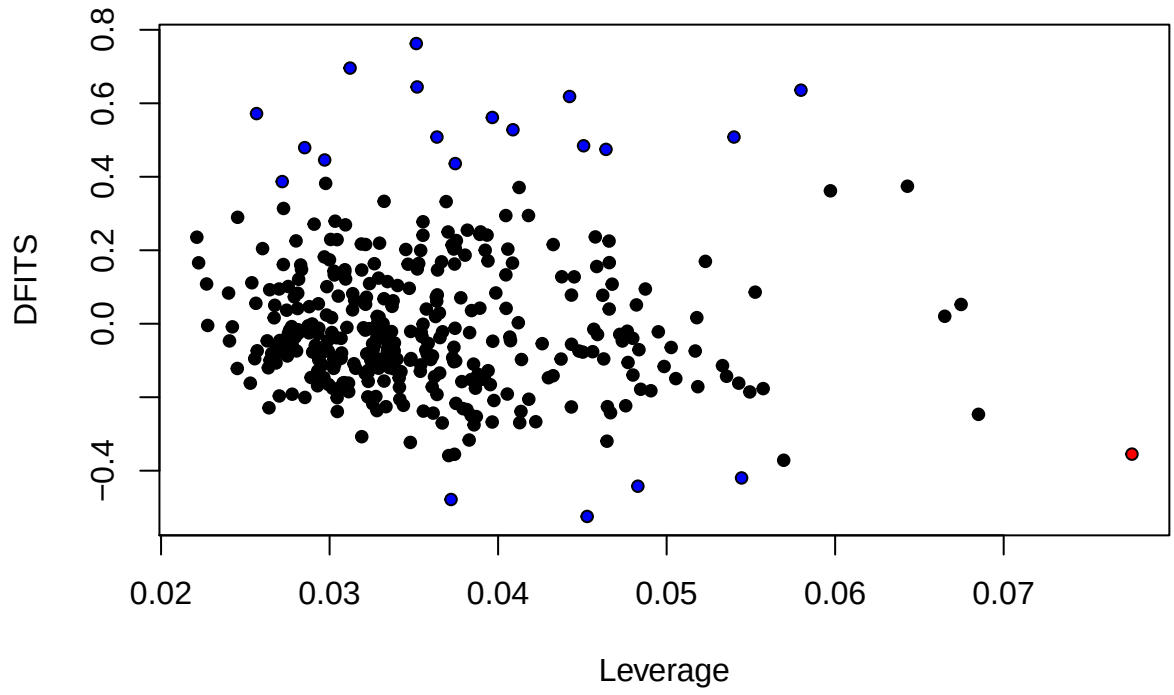



```
## checking outlier
par(mfrow = c(1, 1))

# 1.leverage point
pii = influence(m5)$hat
lev.idx = pii > 2*p/n
color = rep("black", len=n)
color[lev.idx] = 'red'
```

```r
# 2.cooks distance
Ci = cooks.distance(m5)
sum(Ci > qf(0.5,p,n-p)) # none
```

```
## [1] 0
```

```r
# 3.dfits
dfits = dffits(m5)
inf.idx = abs(dfits) >= 2*sqrt(p/(n-p))
color[inf.idx] = 'blue'

# there is no influential point which is highly different from others
plot(pii, dfits, type='n',
     xlab = 'Leverage', ylab = 'DFITS')
points(pii, dfits, pch=21, cex=0.8, bg=color)
```




```r
## model selection
library(olsrr)
```

```
##
## Attaching package: 'olsrr'

## The following object is masked from 'package:datasets':
##
##     rivers
```

```r
# forward selection
mfwd_aic <- ols_step_forward_aic(m5, details=FALSE)
mfwd_aic$model
```

```
##
## Call:
## lm(formula = paste(response, "~", paste(preds, collapse = " + ")),
##     data = l)
##
## Coefficients:
##     (Intercept)  as.factor(day)2  as.factor(day)3  as.factor(day)4
##        22.11469          0.48546          4.64968          3.63300
## as.factor(day)5  as.factor(day)6  as.factor(day)7            maxWS
##         4.87242         13.51101         11.82652         -0.56808
##          maxSun           lowtem
##        -0.90116          0.04846
```

```r
# backward elimination
mbwd_aic <- ols_step_backward_aic(m5, details=FALSE)
mbwd_aic$predictors
```

```
## [1] "avgHum" "tem15"  "avgCl"
```

```r
# stepwise selection
mboth_aic <- ols_step_both_aic(m5, details=FALSE)
mboth_aic$predictors
```

```
## [1] "as.factor(day)" "maxWS"          "maxSun"         "lowtem"
```

```r
# the same results from 3 methods
# drop avgHum,tem15, avgCl

final = mfwd_aic$model
summary(final)
```

```
##
## Call:
## lm(formula = paste(response, "~", paste(preds, collapse = " + ")),
##     data = l)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.933  -3.599  -1.192   3.146  22.269
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     22.11469    1.75158  12.626  < 2e-16 ***
## as.factor(day)2  0.48546    1.12539   0.431  0.66647
## as.factor(day)3  4.64968    1.13427   4.099 5.15e-05 ***
## as.factor(day)4  3.63300    1.13477   3.202  0.00149 **
## as.factor(day)5  4.87242    1.13872   4.279 2.43e-05 ***
```

```
## as.factor(day)6 13.51101    1.14131  11.838  < 2e-16 ***
## as.factor(day)7 11.82652    1.12988  10.467  < 2e-16 ***
## maxWS           -0.56808    0.32038  -1.773  0.07707 .
## maxSun          -0.90116    0.40791  -2.209  0.02781 *
## lowtem           0.04846    0.03145   1.541  0.12424
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.755 on 351 degrees of freedom
## Multiple R-squared:  0.4397, Adjusted R-squared:  0.4253
## F-statistic:  30.6 on 9 and 351 DF,  p-value: < 2.2e-16
```

```
## ANOVA test
final2 = lm(sqrt(aud) ~ maxWS + maxSun + as.factor(day), data=data.rem)

anova(final2, final) # NOT reject
```

```
## Analysis of Variance Table
##
## Model 1: sqrt(aud) ~ maxWS + maxSun + as.factor(day)
## Model 2: sqrt(aud) ~ as.factor(day) + maxWS + maxSun + lowtem
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1    352 11703
## 2    351 11624  1    78.633 2.3744 0.1242
```

```
summary(final2)
```

```
##
## Call:
## lm(formula = sqrt(aud) ~ maxWS + maxSun + as.factor(day), data = data.rem)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.939  -3.669  -1.278   3.272  21.522
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      22.6897     1.7147  13.232  < 2e-16 ***
## maxWS            -0.6871     0.3115  -2.205  0.02807 *
## maxSun           -0.7072     0.3888  -1.819  0.06973 .
## as.factor(day)2   0.4300     1.1270   0.382  0.70300
## as.factor(day)3   4.6362     1.1364   4.080 5.59e-05 ***
## as.factor(day)4   3.5857     1.1366   3.155  0.00174 **
## as.factor(day)5   4.8007     1.1400   4.211 3.23e-05 ***
## as.factor(day)6  13.4532     1.1429  11.771  < 2e-16 ***
## as.factor(day)7  11.8256     1.1321  10.446  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.766 on 352 degrees of freedom
## Multiple R-squared:  0.4359, Adjusted R-squared:  0.4231
## F-statistic:    34 on 8 and 352 DF,  p-value: < 2.2e-16
```

```r
final3 = lm(sqrt(aud) ~ maxWS + as.factor(day), data=data.rem)
anova(final3,final2) # reject
```

```
## Analysis of Variance Table
##
## Model 1: sqrt(aud) ~ maxWS + as.factor(day)
## Model 2: sqrt(aud) ~ maxWS + maxSun + as.factor(day)
##   Res.Df   RSS Df Sum of Sq      F  Pr(>F)
## 1    353 11813
## 2    352 11703  1    110.03 3.3094 0.06973 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Final model ------------------------------------------------------------

summary(final2)
```

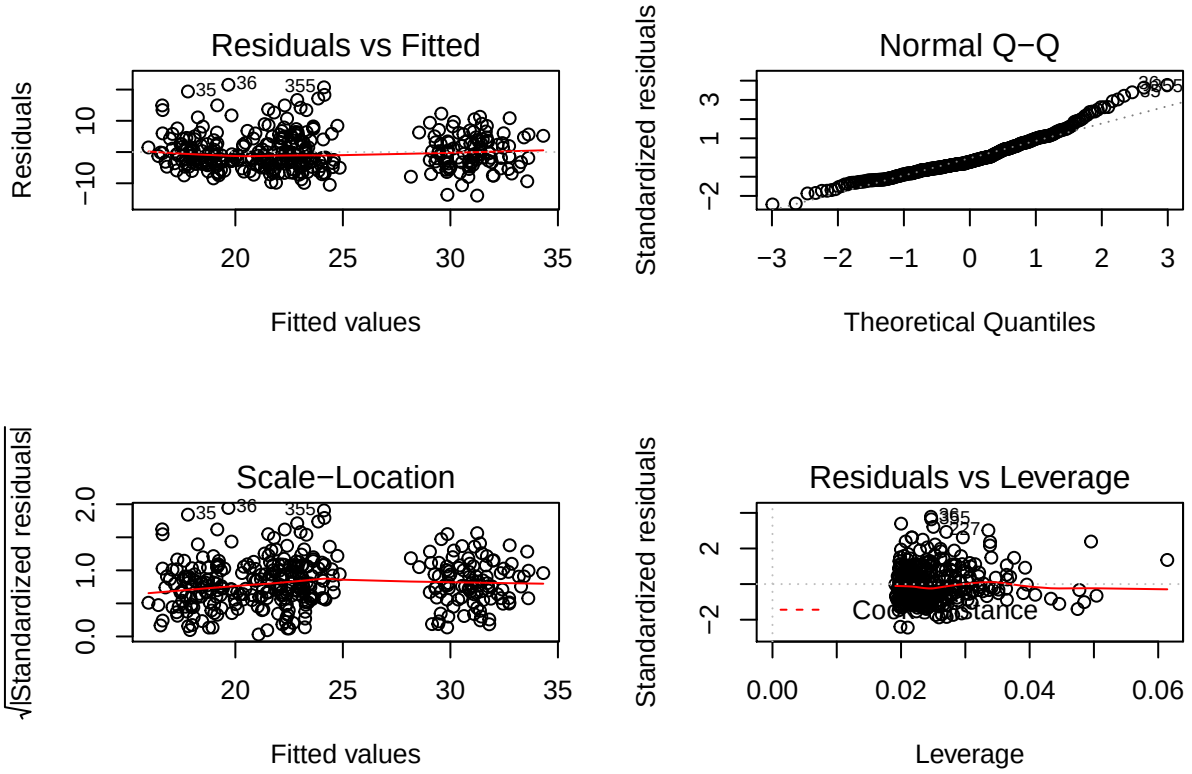
```
##
## Call:
## lm(formula = sqrt(aud) ~ maxWS + maxSun + as.factor(day), data = data.rem)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.939  -3.669  -1.278   3.272  21.522
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      22.6897     1.7147  13.232  < 2e-16 ***
## maxWS            -0.6871     0.3115  -2.205  0.02807 *
## maxSun           -0.7072     0.3888  -1.819  0.06973 .
## as.factor(day)2   0.4300     1.1270   0.382  0.70300
## as.factor(day)3   4.6362     1.1364   4.080 5.59e-05 ***
## as.factor(day)4   3.5857     1.1366   3.155  0.00174 **
## as.factor(day)5   4.8007     1.1400   4.211 3.23e-05 ***
## as.factor(day)6  13.4532     1.1429  11.771  < 2e-16 ***
## as.factor(day)7  11.8256     1.1321  10.446  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.766 on 352 degrees of freedom
## Multiple R-squared:  0.4359, Adjusted R-squared:  0.4231
## F-statistic:    34 on 8 and 352 DF,  p-value: < 2.2e-16
```

```r
# 95% CI
confint(final2)
```

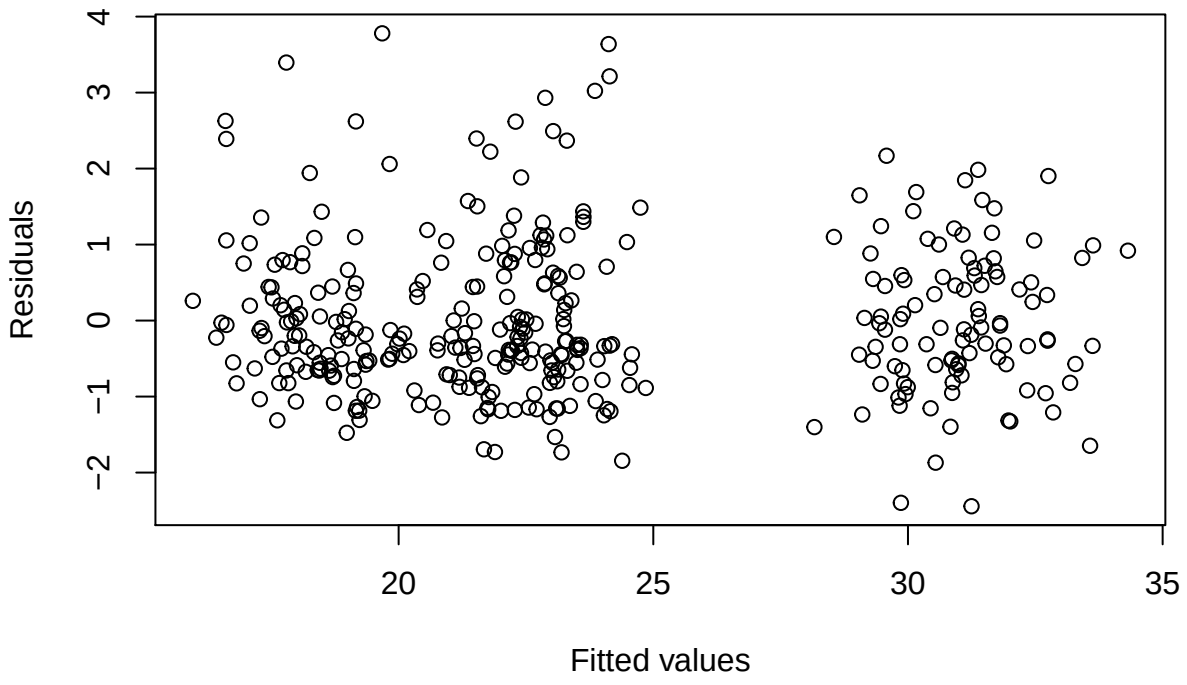
```
##                      2.5 %      97.5 %
## (Intercept)     19.317338 26.06204917
## maxWS           -1.299795 -0.07436347
## maxSun          -1.471855  0.05736223
## as.factor(day)2 -1.786480  2.64656557
## as.factor(day)3  2.401105  6.87126872
```

```
## as.factor(day)4  1.350380  5.82102029
## as.factor(day)5  2.558666  7.04276497
## as.factor(day)6 11.205426 15.70104215
## as.factor(day)7  9.599087 14.05207506
```

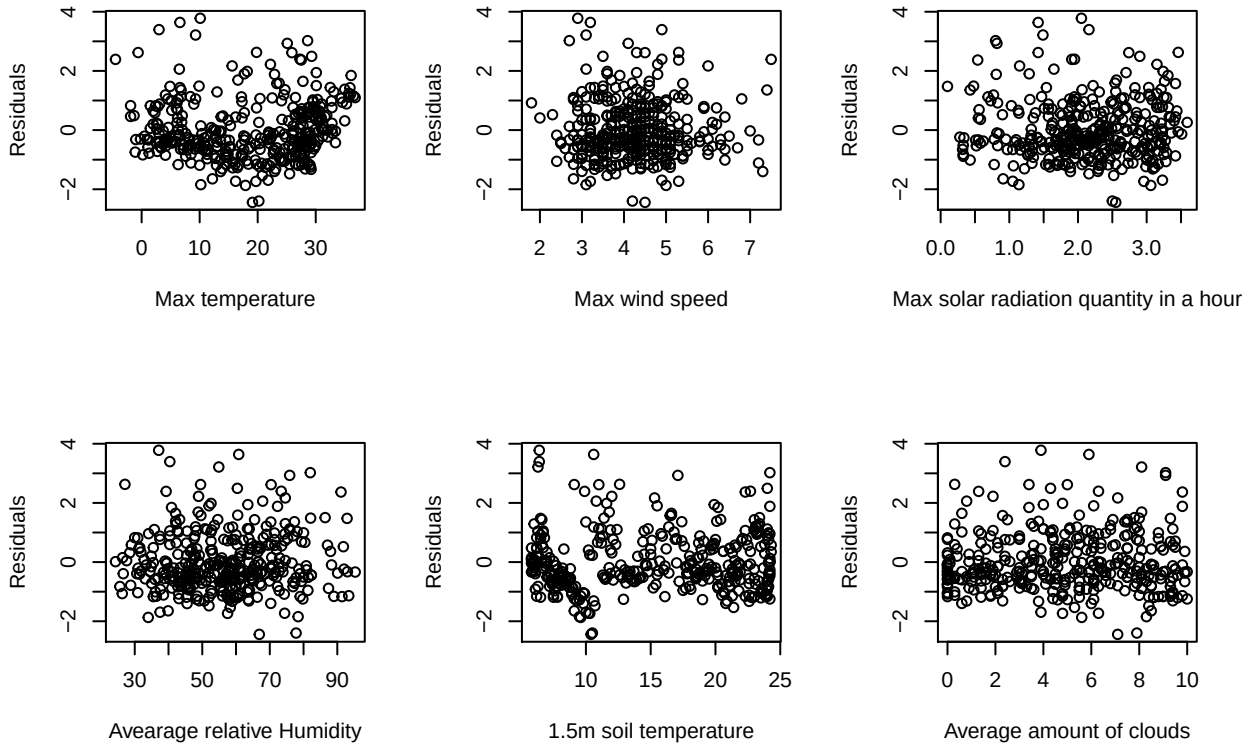
```
X = model.matrix(final2)
n = nrow(X)
p = ncol(X)

## checking assumptions
par(mfrow = c(2, 2))
plot(final2)
```




```
## (y_hat, r)
par(mfrow = c(1, 1))
res = rstandard(final2)
plot(final2$fitted.values, res,
     xlab = 'Fitted values', ylab='Residuals')
```



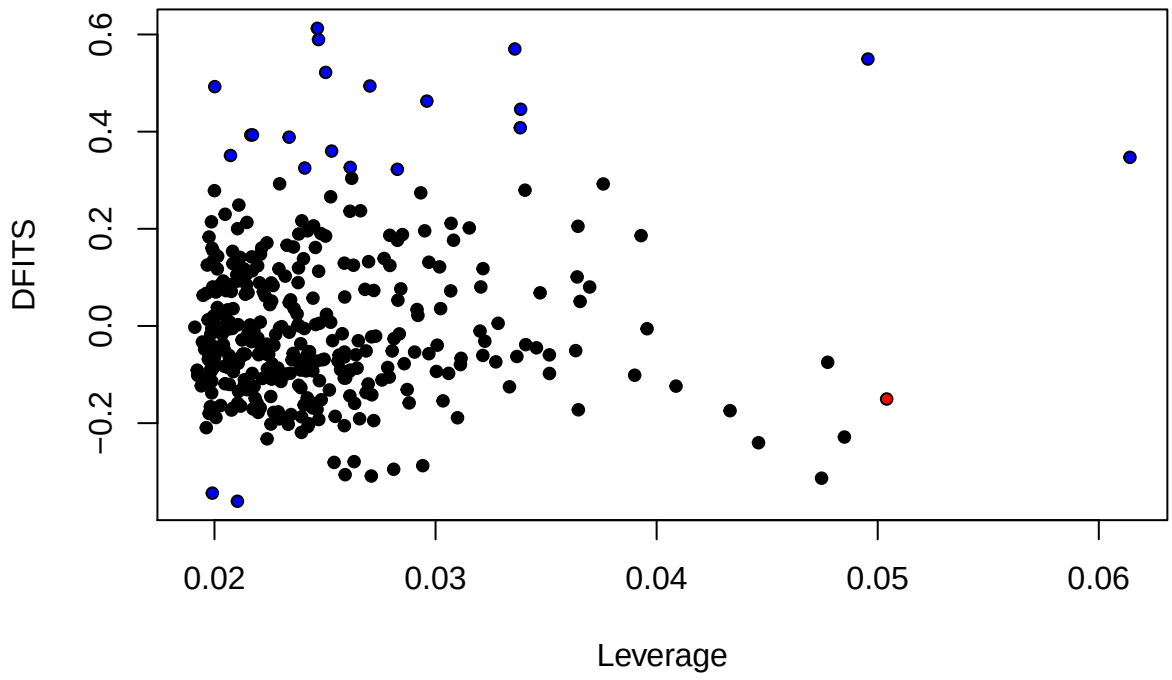

```
## (x, r)
par(mfrow = c(2, 3))
plot(data.rem$hitem, res, xlab="Max temperature", ylab="Residuals")
plot(data.rem$maxWS, res, xlab="Max wind speed", ylab="Residuals")
plot(data.rem$maxSun, res, xlab="Max solar radiation quantity in a hour", ylab="Residuals")
plot(data.rem$avgHum, res, xlab="Avearage relative Humidity", ylab="Residuals")
plot(data.rem$tem15, res, xlab="1.5m soil temperature", ylab="Residuals")
plot(data.rem$avgCl, res, xlab="Average amount of clouds", ylab="Residuals")
```




```
## checking outlier
par(mfrow = c(1, 1))

# 1.leverage point
pii = influence(final2)$hat
lev.idx = pii > 2*p/n
color = rep("black", len=n)
color[lev.idx] = 'red'

# 2.cooks distance
Ci = cooks.distance(final2)
sum(Ci > qf(0.5,p,n-p)) # none
```

```
## [1] 0
```

```
# 3.dfits
dfits = dffits(final2)
inf.idx = abs(dfits) >= 2*sqrt(p/(n-p))
color[inf.idx] = 'blue'

plot(pii, dfits, type='n',
     xlab = 'Leverage', ylab = 'DFITS')
points(pii, dfits, pch=21, cex=0.8, bg=color) # there is no influential point which is highly different
```




```
## multicollinearity
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
vif(final2)
```

```
##                     GVIF Df GVIF^(1/(2*Df))
## maxWS           1.024364  1        1.012109
## maxSun          1.015160  1        1.007552
## as.factor(day) 1.021496  6        1.001774
```

```
## autocorrelation
dwtest(final2, alternative = 'greater')
```

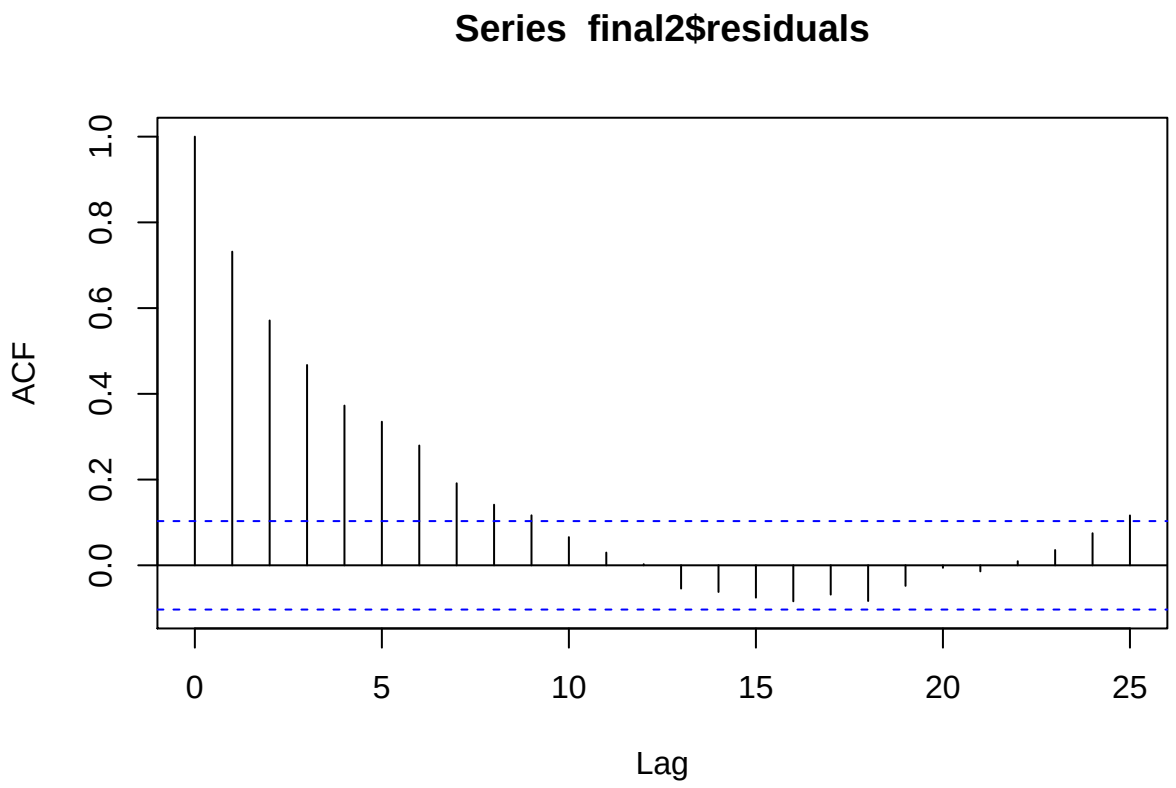
```
##
```

```
##  Durbin-Watson test
##
## data:  final2
## DW = 0.502, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```
runs.test(final2$residuals, alternative="left.sided", plot = FALSE)
```

```
##
##  Runs Test
##
## data:  final2$residuals
## statistic = -12.878, runs = 59, n1 = 180, n2 = 180, n = 360, p-value <
## 2.2e-16
## alternative hypothesis: trend
```

```
acf(final2$residuals, plot=TRUE)
```




```
acf(final2$residuals, plot=FALSE)
```

```
##
## Autocorrelations of series 'final2$residuals', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
```

```
##  1.000  0.732  0.571  0.467  0.373  0.335  0.279  0.191  0.141  0.117  0.066
##     11     12     13     14     15     16     17     18     19     20     21
##  0.030  0.003 -0.054 -0.062 -0.075 -0.084 -0.068 -0.083 -0.048 -0.006 -0.014
##     22     23     24     25
##  0.010  0.036  0.075  0.116
```