Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

Automobile Data Set

Obiettivo:

Realizzare un modello robusto capace di analizzare la variazione del prezzo delle automobili in funzione delle caratteristiche osservate. Utilizziamo come riferimento le informazioni contenute nel dataset Automobile Data Set, il quale ha come fonti:

1) 1985 Model Import Car and Truck Specifications, 1985 Ward's Automotive Yearbook.
2) Personal Auto Manuals, Insurance Services Office, 160 Water Street, New York, NY 10038
3) Insurance Collision Report, Insurance Institute for Highway Safety, Watergate 600, Washington, DC 20037

Analisi preliminare:

Il dataset è composto da specifiche riguardanti l'automobile di tipo generale come il produttore (***make***), il numero di porte (***numOfDoors***) eccetera (***fuelType, aspiration, bodyStyle, driveWheels, engineLocation***) e da informazioni più tecniche come le dimensioni del veicolo (***wheelBase, length, width, height, curbWeight***), le caratteristiche tecniche del motore (***engineType, Cylinders, engineSize, fuelSystem, bore, stroke, compressionRatio, horsepower, peakRpm***) e dettagli sui consumi (***cityMpg, highwayMpg***).
Le restanti variabili ***symboling*** e ***normalizedLosses*** rappresentano rispettivamente il grado di rischio dell'auto rispetto al prezzo e la perdita economica media per veicolo assicurato, valore normalizzato in base alle dimensioni del veicolo.

Analizziamo come si distribuisce la variabile target ***price***:

```
Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
5118    7775   10295   13207   16500   45400       4
```

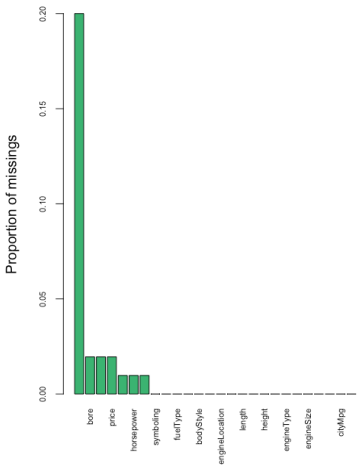
Come prevedibile, essendo una variabile di tipo prezzo, non si distribuisce propriamente come una normale, presenta una coda di destra molto allungata. Notiamo inoltre la presenza di valori mancanti, risolveremo la questione al prossimo step.

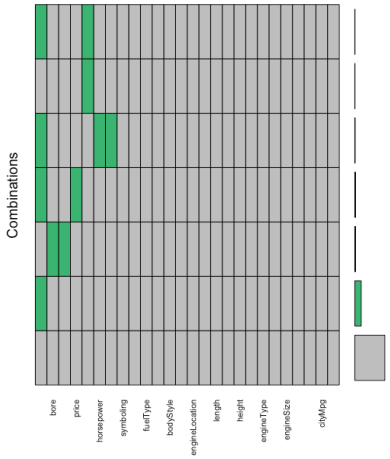
Valori mancanti e imputazione:

Osserviamo numericamente e graficamente come sono distribuiti i valori mancanti all'interno del dataset di riferimento, notiamo come la mancanza di osservazioni coinvolga sette variabili tra cui il target, quest'ultimo con una percentuale di missing values prossima al 2%.
La variabile *normalizedLosses* raggiunge la soglia del 20% di valori mancanti, decidiamo quindi di non considerarla per le analisi, riguardo al target decidiamo di rimuovere le 4 osservazioni in cui è assente. Le rimanenti 5 variabili caratterizzate da valori nulli vengono elaborate tramite la libreria Mice Imputation, quest'ultima permette di ricavare i valori mancanti utilizzando rispettivamente un metodo predictive mean matching per le variabili numeriche e una regressione logistica per la variabile binaria *(numOfDoors)*.

Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

```
Variables sorted by number of missings:
        Variable       Count
normalizedLosses 0.200000000
            bore 0.019512195
          stroke 0.019512195
           price 0.019512195
       numOfDoors 0.009756098
      horsepower 0.009756098
         peakRpm 0.009756098
       symboling 0.000000000
            make 0.000000000
        fuelType 0.000000000
      aspiration 0.000000000
       bodyStyle 0.000000000
```





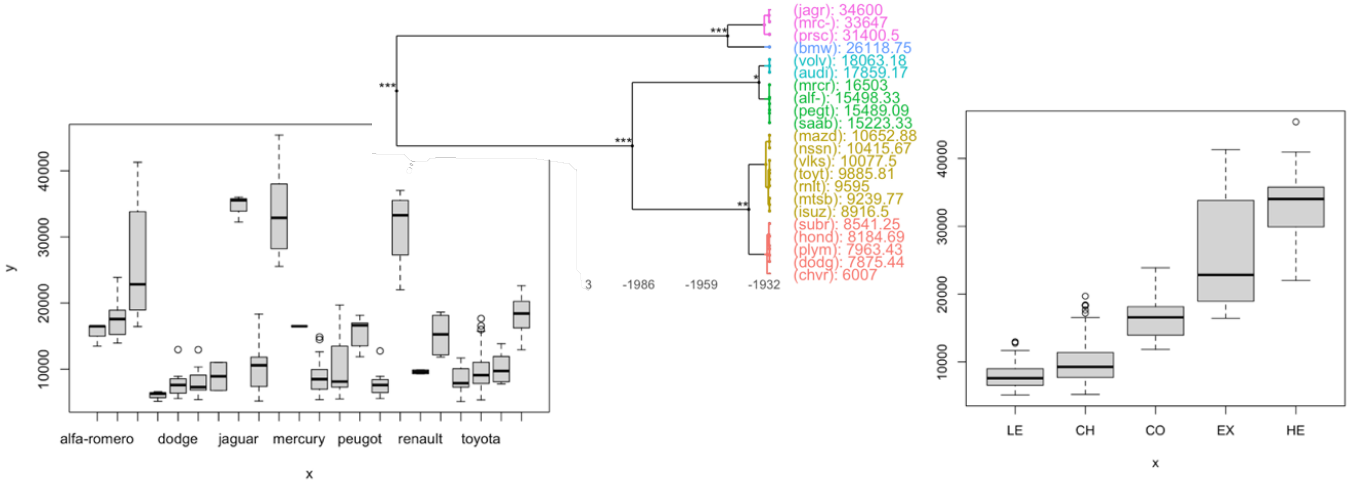
A seguito dell'imputazione ricaviamo un dataset con 201 osservazioni e 25 variabili, disponendo ora di un dataset completo potremmo iniziare a fittare un modello, notiamo tuttavia che molteplici variabili categoriali sono caratterizzate da un elevato numero di livelli, decidiamo quindi di rimandare l'interpolazione del modello di partenza dopo l'optimal grouping.

Nuove variabili e gruppi ottimali:

Iniziamo generando una nuova variabile *volume* calcolata tramite le variabili *lenght, width* e *height*, vedremo in seguito se sarà utile a riassumerle. Modifichiamo inoltre la variabile *engineSize* moltiplicandola per il coefficiente 16.39, in questo modo convertiamo l'unità di misura da CID (cubic inches) a CC (cubic centimeter), misura più utilizzata e di più facile interpretazione.

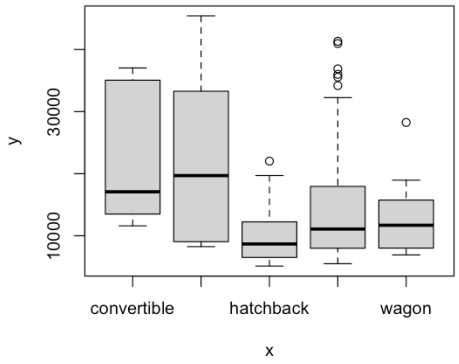
Eseguiamo ora optimal grouping supervisionato sulla variabile symboling, avendo una distribuzione particolare la libreria factor merger non ci forniva un risultato soddisfacente, aggreghiamo quindi manualmente i sei livelli in tre (safe, normal, risky) mantenendone il significato corretto.
La libreria citata in precedenza esegue invece un ottimo lavoro sulle rimanenti variabili categoriali con un numero troppo elevato di livelli (optimal grouping non supervisionato), come possiamo vedere graficamente riduce la variabile make da 22 livelli ad appena 6, decidiamo poi di unirne due manualmente ed otteniamo infine cinque livelli: Low-End, CHeap, COmpetitive, EXpensive, High-End. Analogamente elaboriamo le rimanenti variabili categoriali.

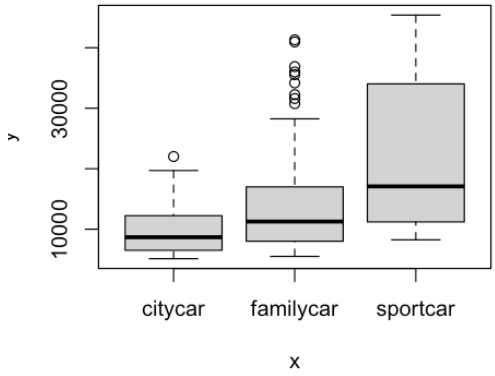
Make:



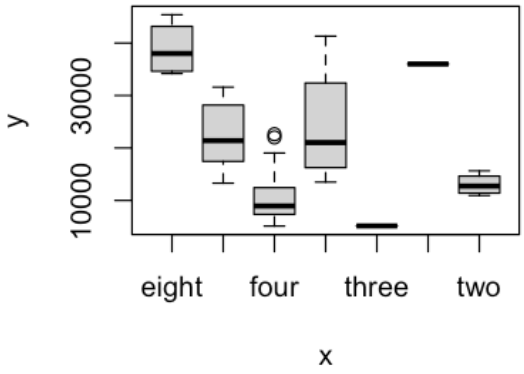
Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

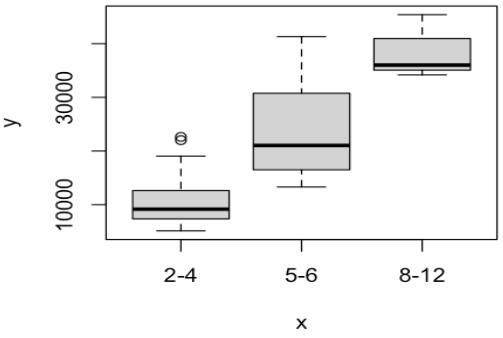
Bodystyle:



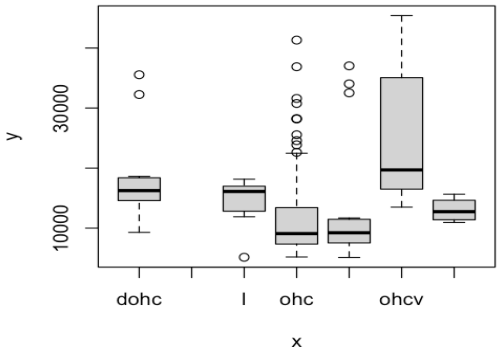


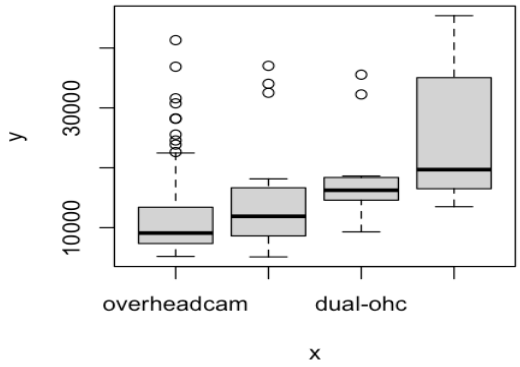
Cylinders:



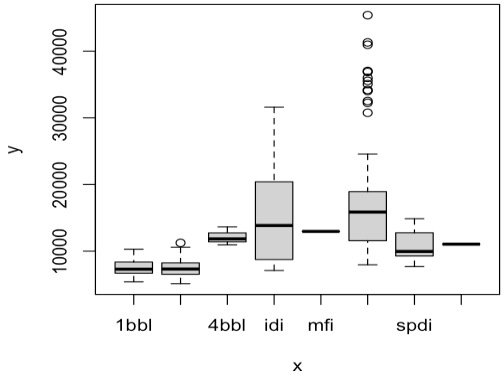


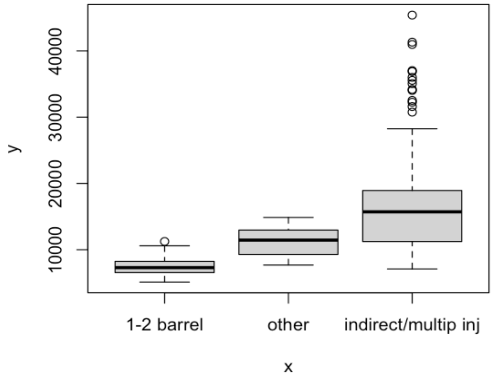
EngineType:





FuelSystem:

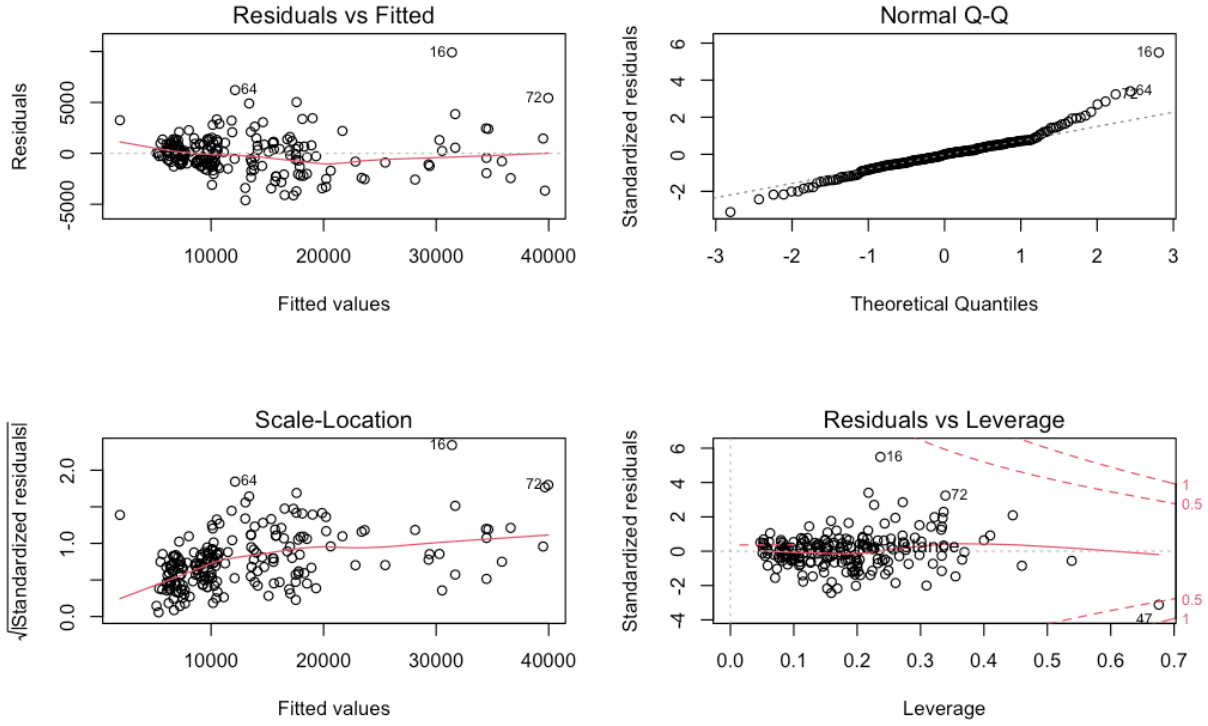




Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

Modello di partenza:

Possiamo ora interpolare il modello completo di partenza da tenere come riferimento




```
Residual standard error: 2065 on 165 degrees of freedom
Multiple R-squared:  0.9443,    Adjusted R-squared:  0.9325
F-statistic: 79.93 on 35 and 165 DF,  p-value: < 2.2e-16
```

Il grafico dei residui vs interpolati evidenzia la necessità di trasformare la y, il qqplot conferma la non normalità osservata in precedenza e la rispettiva coda destra allungata. Il grafico Scale-Location suggerisce la presenza di eteroschedasticità nel modello, infine riscontriamo la presenza di almeno un punto influente e diversi outliers. Tutti questi problemi verranno trattati nei punti succesivi.

Collinearità:

Variabili qualitative:

	X1	Row	Column	Chi.Square	df	p.value	n	u1	u2	nMinu1u2	Chi.Square.norm
1	1	symboling	make	25.735	NA	0.00250	201	2	4	402	0.064018041
2	2	symboling	fuelType	4.389	NA	0.09345	201	2	1	201	0.021835810
3	3	symboling	aspiration	6.511	NA	0.04998	201	2	1	201	0.032391439
4	4	symboling	numOfDoors	63.726	NA	0.00050	201	2	1	201	0.317042443
22	22	fuelType	bodyStyle	9.158	NA	0.01749	201	1	2	201	0.045562507
23	23	fuelType	driveWheels	3.457	NA	0.13693	201	1	2	201	0.017197397
24	24	fuelType	engineLocation	0.337	NA	1.00000	201	1	1	201	0.001674201
25	25	fuelType	engineType	4.108	NA	0.23488	201	1	3	201	0.020440145
26	26	fuelType	Cylinders	1.497	NA	0.40030	201	1	2	201	0.007447736
52	52	engineLocation	fuelSystem	2.420	NA	0.42729	201	1	2	201	0.012040043
53	53	engineType	Cylinders	111.942	NA	0.00050	201	3	2	402	0.278463281
54	54	engineType	fuelSystem	25.658	NA	0.00100	201	3	2	402	0.063825838
55	55	Cylinders	fuelSystem	38.452	NA	0.00050	201	2	2	402	0.095651455

Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

Inizialmente rileviamo tanti problemi visto l'elevato numero di variabili categoriali, eseguiamo i seguenti passaggi intermedi per ottenere la configurazione ottimale:

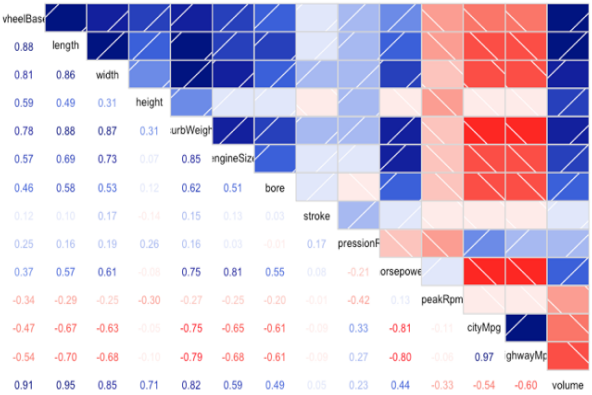
- Togliamo *make*, presenta problemi con tutte le variabili (Chi.Square elevato e p.value basso)
- Togliamo *symboling*, collineare con quasi tutte le variabili
- Togliamo *engineLocation* e *bodyStyle* poichè risultano ancora molto relazionate con *numOfDoors*
- Togliamo *fuelType* e *fuelSystem* poiché sembrano essere spiegate da *aspiration*
- *EngineType*, *driveWheels* e *Cylinders* risultano molto relazionate tra loro, dopo aver provato diverse configurazioni decidiamo di tenere solamente *Cylinders*

Otteniamo:

X1	Row	Column	Chi.Square	df	p.value	n	u1	u2	nMinu1u2	Chi.Square.norm
1	aspiration	numOfDoors	0.798	NA	0.44378	201	1	1	201	0.003971106
2	aspiration	Cylinders	1.257	NA	0.60670	201	1	2	201	0.006252430
3	numOfDoors	Cylinders	0.905	NA	0.69115	201	1	2	201	0.004501301

Variabili quantitative:

	VIF	TOL	Wi	Fi	Leamer	CVIF
wheelBase	8.2867	0.1207	104.8164	114.1583	0.3474	-0.2815
length	195.0660	0.0051	2791.5641	3040.3666	0.0716	-6.6267
width	53.9547	0.0185	761.7327	829.6233	0.1361	-1.8329
height	83.0526	0.0120	1180.2952	1285.4908	0.1097	-2.8214
curbWeight	16.2762	0.0614	219.7425	239.3273	0.2479	-0.5529
engineSize	7.0081	0.1427	86.4235	94.1261	0.3777	-0.2381
bore	1.9535	0.5119	13.7155	14.9380	0.7155	-0.0664
stroke	1.2012	0.8325	2.8943	3.1523	0.9124	-0.0408
compressionRatio	2.2512	0.4442	17.9974	19.6015	0.6665	-0.0765
horsepower	8.2698	0.1209	104.5727	113.8930	0.3477	-0.2809
peakRpm	2.1854	0.4576	17.0520	18.5718	0.6764	-0.0742
cityMpg	26.9919	0.0370	373.8832	407.2061	0.1925	-0.9170
highwayMpg	24.8227	0.0403	342.6797	373.2216	0.2007	-0.8433
volume	676.6945	0.0015	9719.6055	10585.8805	0.0384	-22.9884



- Iniziamo a togliere le variabili che abbiamo intenzionalmente riassunto con *volume*, per lo stesso motivo togliamo anche *wheelBase*
- Togliamo le variabili relative ai consumi poichè spiegate da *Horsepower*
- Togliamo anche *engineSize* poiché riassunta da *horsepower*, *bore* e *stroke*
- Nonostante *curbWeight* sia ancora leggermente sopra la soglia limite decidiamo di tenerla per il momento

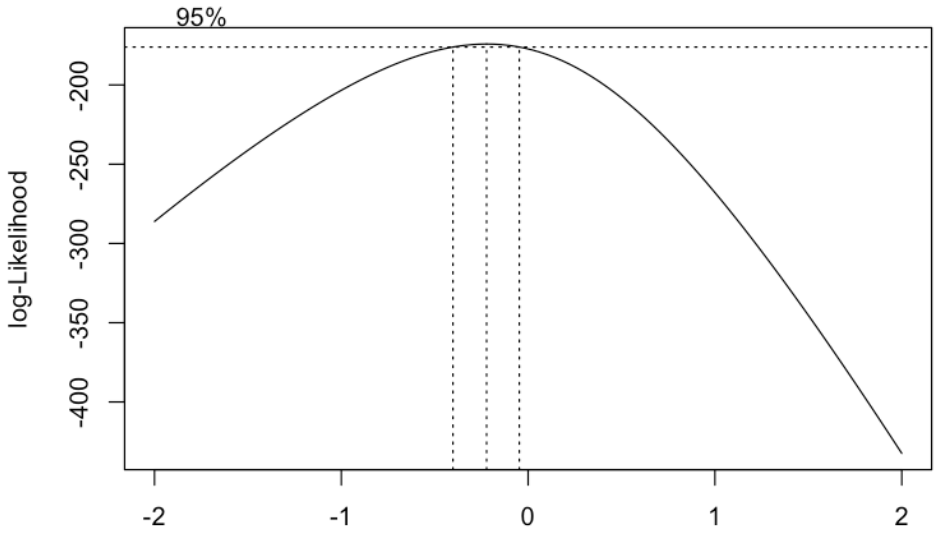
	VIF	TOL	Wi	Fi	Leamer	CVIF	Klein	IND1	IND2
curbWeight	9.4827	0.1055	274.2733	330.8245	0.3247	-1.9667	1	0.0033	1.7291
bore	1.7930	0.5577	25.6404	30.9270	0.7468	-0.3719	0	0.0172	0.8549
stroke	1.1667	0.8571	5.3890	6.5001	0.9258	-0.2420	0	0.0265	0.2761
compressionRatio	1.4835	0.6741	15.6321	18.8552	0.8210	-0.3077	0	0.0208	0.6299
horsepower	4.2262	0.2366	104.3134	125.8213	0.4864	-0.8765	0	0.0073	1.4755
peakRpm	1.4304	0.6991	13.9147	16.7837	0.8361	-0.2966	0	0.0216	0.5816
volume	4.0265	0.2484	97.8582	118.0352	0.4983	-0.8351	0	0.0077	1.4529

Provando a interpolare le covariate rimanenti si ottiene un leggero miglioramento, soprattutto riguardante i punti influenti, si evidenzia però ulteriormente la necessità di eseguire una trasformazione lineare per il target.

Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

Trasformazioni:

Box-cox, trasformazione del target:



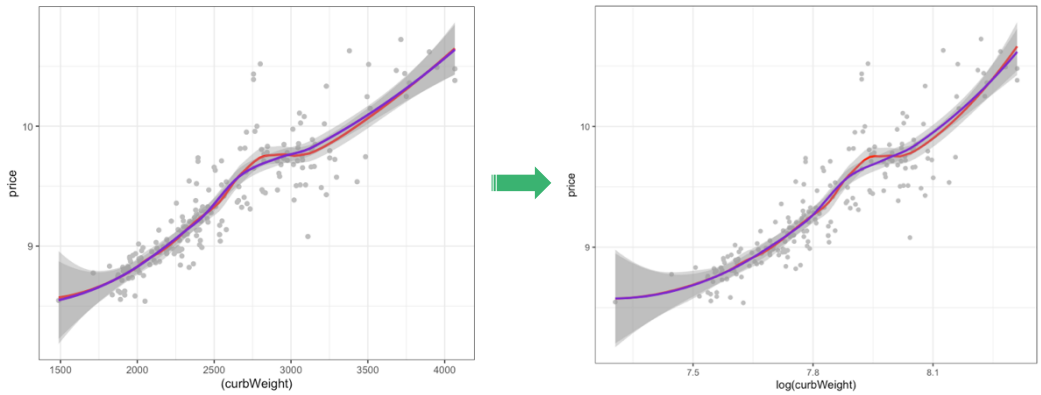
La lambda risulta essere pari a -0.22, decidiamo di trasformare il target utilizzando il logaritmo, il modello risulta evidentemente migliorato com'era prevedibile essendo la variabile di tipo prezzo, il logaritmo aiuta infatti a comprimere la coda di destra.

Gam, Trasformazioni delle covariate:

Utilizzando il generalized additive model possiamo rappresentare la relazione tra target e covariate tramite funzioni smooth, è utile per capire quali covariate non sono ben approssimate da una funzione lineare e possono essere migliorate.

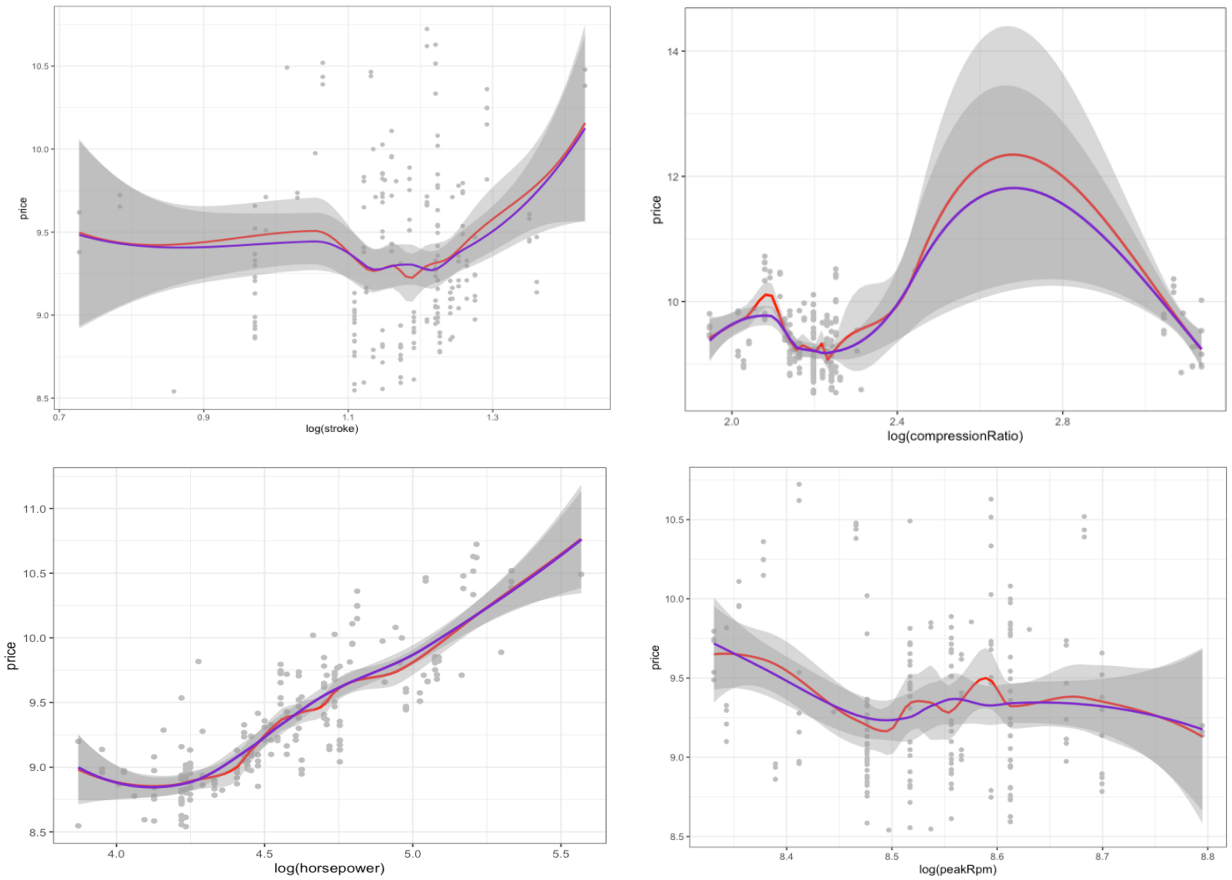
	Npar Df	Npar F	Pr(F)	
(Intercept)				
aspiration				
numOfDoors				
Cylinders				
s(curbWeight)	3	3.5668	0.015426	*
s(bore)	3	1.2017	0.310846	
s(stroke)	3	3.0240	0.031187	*
s(compressionRatio)	3	4.4631	0.004812	**
s(horsepower)	3	3.1543	0.026348	*
s(peakRpm)	3	10.5503	2.151e-06	***
s(volume)	3	1.4391	0.233232	

Curbweight, trasformazione logaritmica (rappresentazioni con span 0.5 e 0.7):

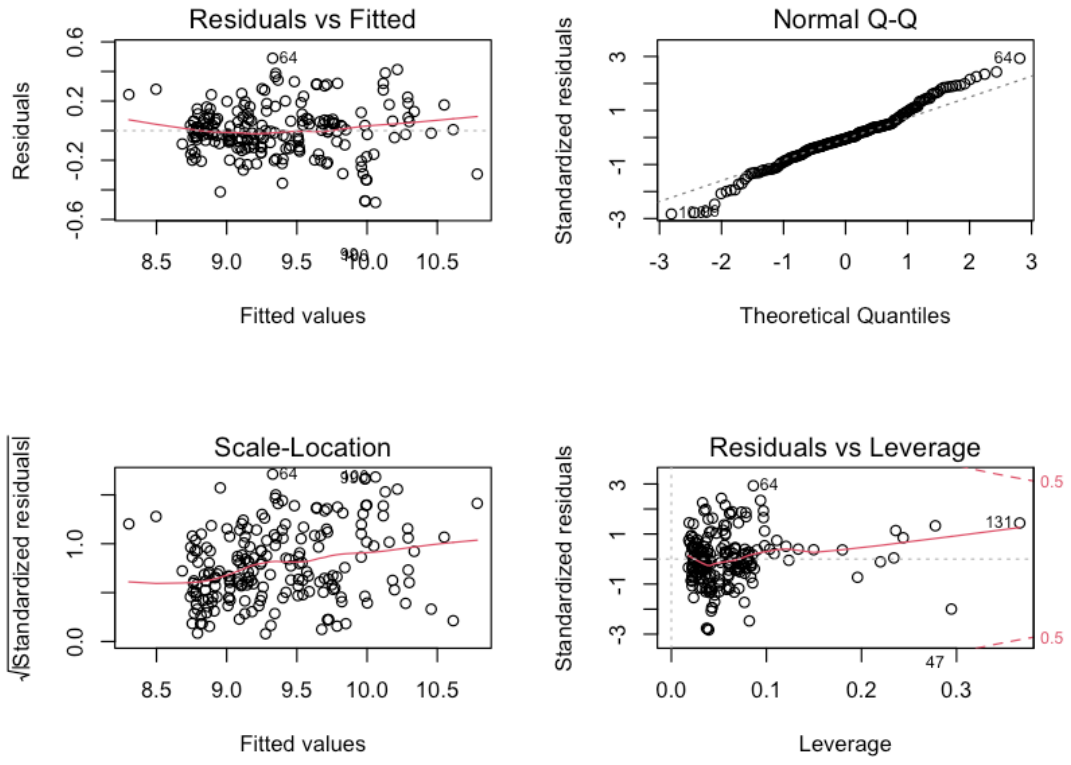


Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

Analogamente vediamo le trasformazioni di stroke, compressionRatio, horsepower e peakRpm:



Nonostante le trasformazioni, non tutte le covariate possono essere approssimate correttamente da una funzione lineare, tuttavia possiamo già vedere miglioramenti significativi attraverso la diagnostica del modello.



Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

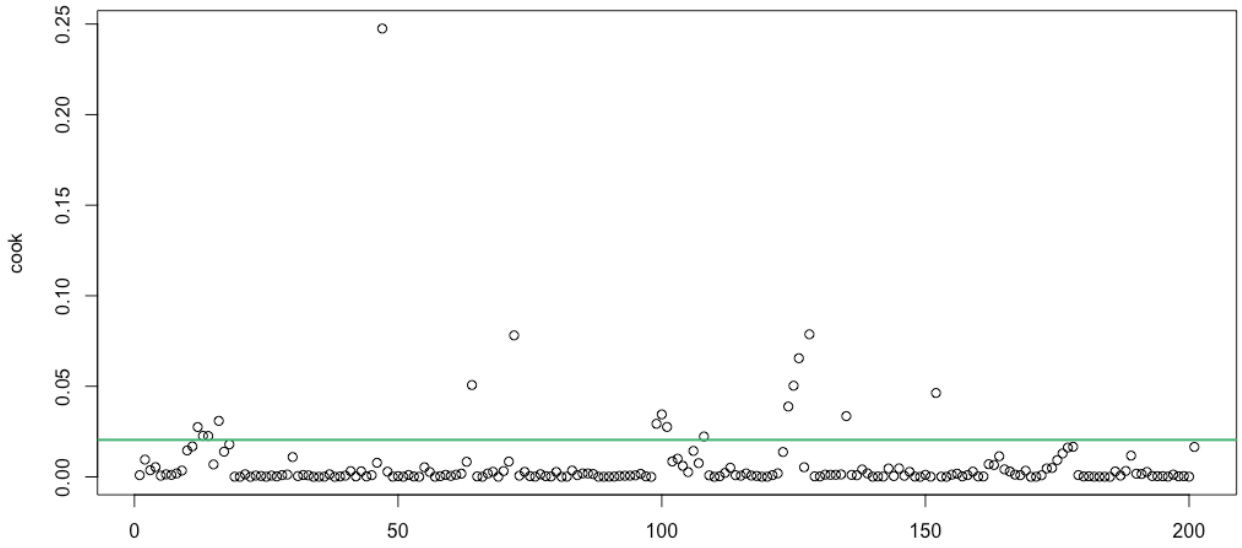
Selezione del modello:

Per la selezione delle covariate da mantenere nel modello utiliziamo i metodi AIC e SBC, decidiamo di utilizzare il modello suggerito dallo shwarz bayesian criterior poiché penalizza meglio i modelli con un alto numero di covariate. Tramite un test Chi-Square confermiamo inoltre la convenienza del modello scelto.

```
lm(formula = price ~ Cylinders + log(curbWeight) + log(stroke) +
    log(compressionRatio) + log(horsepower) + log(peakRpm), data = data_trasf)

Residuals:
     Min       1Q   Median       3Q      Max
-0.50471 -0.09153 -0.01721  0.09414  0.46837

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)          -6.90188    1.76687  -3.906  0.00013 ***
Cylinders5-6          0.19482    0.04354   4.474 1.31e-05 ***
Cylinders8-12         0.41200    0.09132   4.512 1.12e-05 ***
log(curbWeight)       1.34106    0.12912  10.386  < 2e-16 ***
log(stroke)          -0.23616    0.11733  -2.013  0.04553 *
log(compressionRatio)  0.23200    0.05624   4.125 5.50e-05 ***
log(horsepower)       0.49624    0.07716   6.432 9.70e-10 ***
log(peakRpm)          0.37404    0.15653   2.390  0.01783 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1777 on 193 degrees of freedom
Multiple R-squared:  0.8792,    Adjusted R-squared:  0.8749
F-statistic: 200.7 on 7 and 193 DF,  p-value: < 2.2e-16
```

```
  Res.Df    RSS Df Sum of Sq Pr(>Chi)
1    189 5.7723
2    193 6.0926 -4  -0.32029  0.03297 *
```

Outliers:



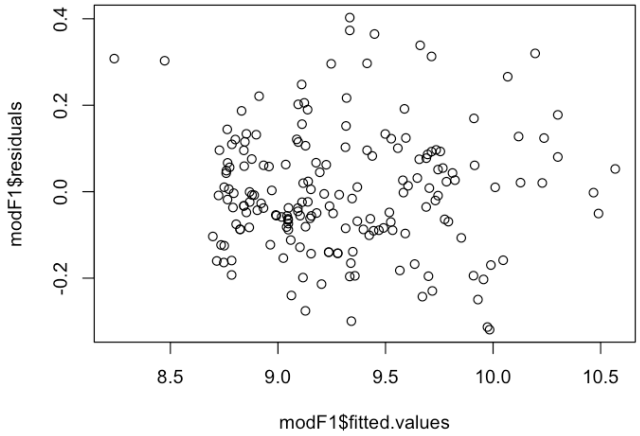
Ricaviamo la soglia di Cook con la formula convenzionale 4/(nrow-npar), i punti influenti rappresentano l’8.46% dei dati, decidiamo quindi di rimuoverli.

Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

Eteroschedasticità:

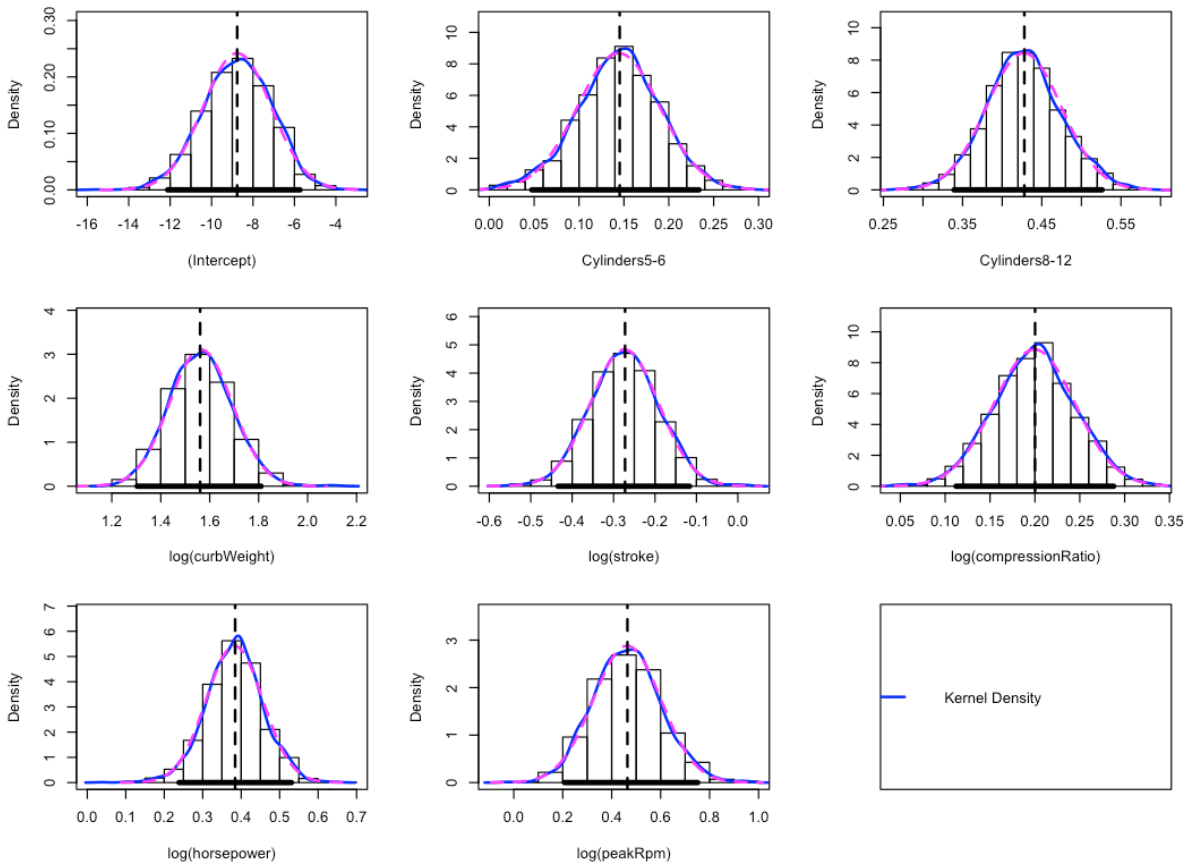
```
        studentized Breusch-Pagan test

data:  modF1
BP = 10.161, df = 7, p-value = 0.1796
```

OK: Error variance appears to be homoscedastic (p = 0.104).



Il p-value del Breush-Pagan test è superiore del 5%, cadiamo quindi nella regione di accettazione dell'ipotesi nulla, il nostro modello risulta essere omoschedastico

Bootstrap:



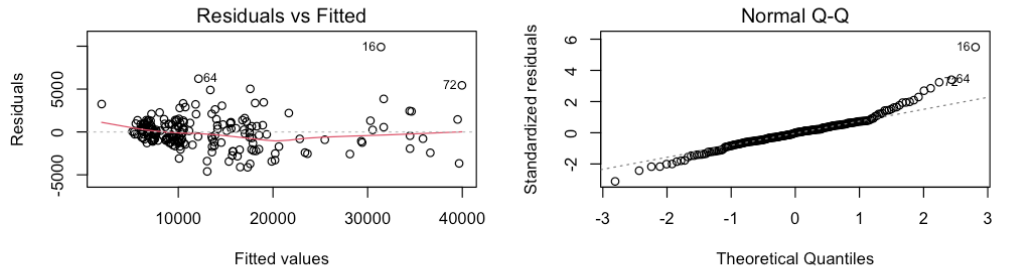
Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

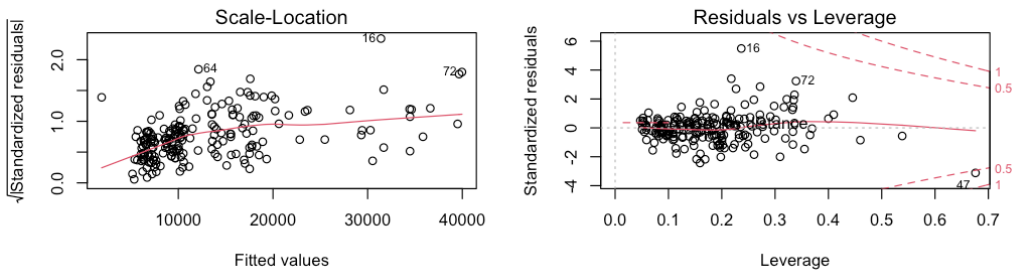
	Estimate	2.5 %	97.5 %
(Intercept)	-8.7648539	-12.07307380	-5.6981889
Cylinders5-6	0.1452708	0.04864607	0.2344623
Cylinders8-12	0.4279910	0.33556937	0.5228739
log(curbWeight)	1.5609400	1.31503450	1.8205873
log(stroke)	-0.2722377	-0.43386998	-0.1166040
log(compressionRatio)	0.2002042	0.11129965	0.2867665
log(horsepower)	0.3843642	0.23504606	0.5287481
log(peakRpm)	0.4643408	0.20183783	0.7441912

Come possiamo vedere, nessun coefficiente assume valore nullo nell'intervallo di confidenza, confermiamo quindi la scelta di tenere tutte le covariate nel modello. Avendo quindi ottenuto il modello finale, possiamo ora verificare i miglioramenti riscontrati rispetto al modello di partenza.

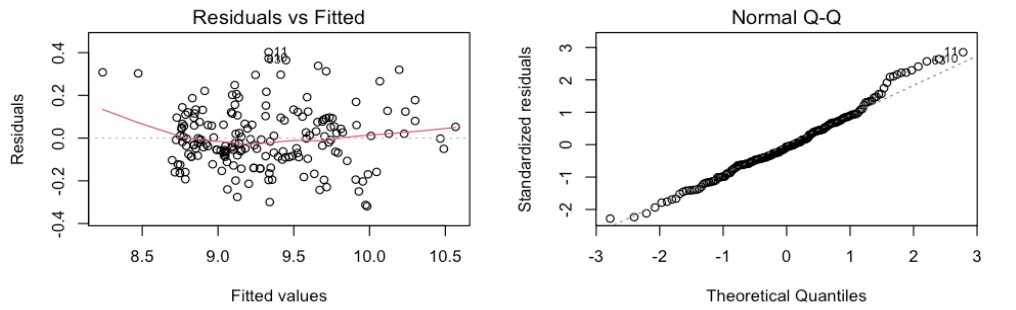
Confronto modelli:

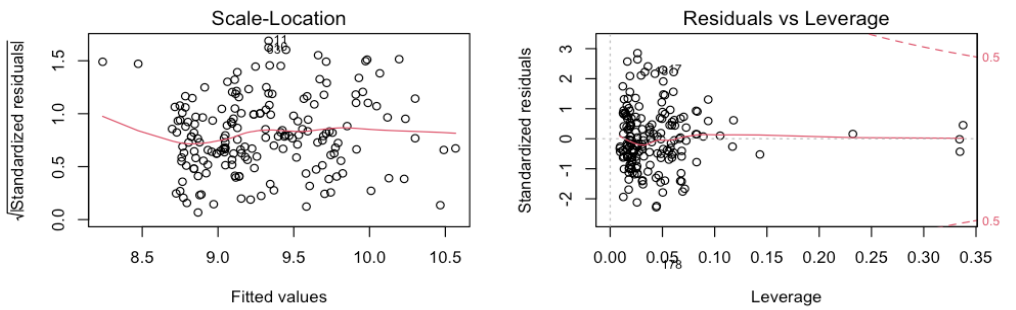
Diagnostica modello iniziale





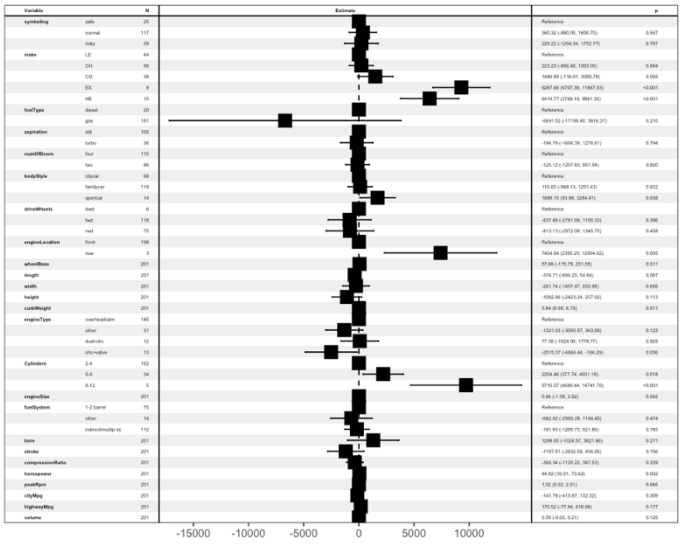
Diagnostica modello finale



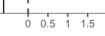


Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

Libreria Forest model:



Variable		N	Estimate	p
Cylinders	2-4	157		Reference
	5-6	24		0.15 (0.07, 0.22)	<0.001
	8-12	3		0.43 (0.25, 0.61)	<0.001
log(curbWeight)		184		1.56 (1.31, 1.81)	<0.001
log(stroke)		184		-0.27 (-0.47, -0.07)	0.007
log(compressionRatio)		184		0.20 (0.10, 0.30)	<0.001
log(horsepower)		184		0.38 (0.23, 0.54)	<0.001
log(peakRpm)		184		0.46 (0.19, 0.74)	0.001



Summary modello finale:

```
Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        -8.76485    1.63234  -5.369 2.47e-07 ***
Cylinders5-6        0.14527    0.03945   3.683 0.000307 ***
Cylinders8-12       0.42799    0.09021   4.744 4.31e-06 ***
log(curbWeight)     1.56094    0.12613  12.375  < 2e-16 ***
log(stroke)        -0.27224    0.10030  -2.714 0.007305 **
log(compressionRatio)  0.20020    0.05083   3.939 0.000118 ***
log(horsepower)     0.38436    0.07767   4.949 1.74e-06 ***
log(peakRpm)        0.46434    0.14149   3.282 0.001244 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1431 on 176 degrees of freedom
Multiple R-squared:  0.908,     Adjusted R-squared:  0.9043
F-statistic: 248.1 on 7 and 176 DF,  p-value: < 2.2e-16
```

Verifica assunzioni (libreria performance):

```
                  Value  p-value                Decision
Global Stat      13.8480 0.007796 Assumptions NOT satisfied!
Skewness          5.5291 0.018703 Assumptions NOT satisfied!
Kurtosis          0.4659 0.494867    Assumptions acceptable.
Link Function     6.7863 0.009186 Assumptions NOT satisfied!
Heteroscedasticity 1.0667 0.301690    Assumptions acceptable.
```

Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

Modello logistico:

Come target del modello logistico decidiamo di tenere la variabile prezzo. Osservando la distribuzione decidiamo di eseguire un cut in prossimità del terzo quartile (≈ 15700) e distinguere le automobili in commerciali e costose:

```
Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
5151    7669    9984   12306   15705   40960
```

```
  0   1
138  46
```

Ricaviamo il seguente generalized linear model:

```
glm(formula = price ~ Cylinders + log(curbWeight) + log(stroke) +
    log(compressionRatio) + log(horsepower) + log(peakRpm), family = "binomial",
    data = data_Logistic)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.66737  -0.16510  -0.03606  -0.00069   2.28103

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)        -220.7900    70.4573  -3.134 0.001726 **
Cylinders5-6          1.1579     0.9257   1.251 0.210976
Cylinders8-12        13.2642  2268.0788   0.006 0.995334
log(curbWeight)      15.2211     4.1514   3.666 0.000246 ***
log(stroke)          -3.0519     2.4305  -1.256 0.209229
log(compressionRatio)  2.9986     1.5660   1.915 0.055517 .
log(horsepower)        4.9621     2.1150   2.346 0.018967 *
log(peakRpm)           8.4874     5.3397   1.590 0.111947
```

Poche variabili appaiono significative secondo il Wald chi-square test. Proviamo quindi ad eseguire model selection per verificare se è possibile togliere alcune covariate. Il criterio di Akaike ci suggerisce di tenere solo quattro covariate, vediamo il modello suggerito:

```
glm(formula = price ~ log(curbWeight) + log(compressionRatio) +
    log(horsepower) + log(peakRpm), family = "binomial", data = data_Logistic)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.38740  -0.19471  -0.03021   0.01471   2.39059

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)        -267.750     66.793  -4.009 6.11e-05 ***
log(curbWeight)      17.045      4.061   4.197 2.71e-05 ***
log(compressionRatio)  3.136      1.514   2.071   0.0383 *
log(horsepower)        4.503      1.840   2.447   0.0144 *
log(peakRpm)          12.113      4.853   2.496   0.0126 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Wald

Davide Vettore 868855
Federico Mammina 869411
Lorenzo Ferrari 866372

```
Model:
price ~ log(curbWeight) + log(compressionRatio) + log(horsepower) +
    log(peakRpm)
                     Df Deviance     AIC     LRT  Pr(>Chi)
<none>                    68.930  78.930
log(curbWeight)       1   93.701 101.701 24.7716 6.454e-07 ***
log(compressionRatio) 1   73.625  81.625  4.6957  0.030239 *
log(horsepower)       1   75.094  83.094  6.1644  0.013035 *
log(peakRpm)          1   75.798  83.798  6.8683  0.008774 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

LRT

Decidiamo di utilizzare il modello suggerito, possiamo ora verificarne l'accuratezza:

```
          predicted
observed   0   1
       0 132   6
       1   7  39
```

```
          predicted
observed          0          1
       0 0.71739130 0.03260870
       1 0.03804348 0.21195652
> accuracy = sum(diag(prob))
> accuracy
[1] 0.9293478
```

Il modello è stato in grado di prevedere correttamente il 92.9% delle osservazioni presenti nel dataset. Calcoliamo infine gli odds ratio, misura di associazione tra le variabili indipendenti e la dipendente, nella pratica corrisponde all'esponenziale dei coefficienti.

```
                            OR         2.5 %       97.5 %
(Intercept)       5.218751e-117 4.999188e-181 7.599398e-66
log(curbWeight)    2.525746e+07  1.772272e+04 1.917180e+11
log(compressionRatio) 2.300693e+01  1.352308e+00 5.503620e+02
log(horsepower)    9.032776e+01  3.122100e+00 4.523671e+03
log(peakRpm)       1.822640e+05  1.921407e+01 4.630982e+09
```

Libreria Forest model: