Linear Regression: MPG in the Motor Trend Cars Dataset

Paul Clark

March 5, 2017

1 Executive Summary

For its 1974 edition, US magazine *Motor Trend* has asked two questions to be addressed using data on fuel consumption and 10 aspects of automobile design and performance for 32 '73-'74 model autos.
Questions:

1. "Is an automatic or manual transmission better for MPG?"
2. "Quantify the MPG-difference between automatic and manual transmissions."

Variables:

```
 [1] "mpg  Miles/(US) gallon"          "cyl  Num cylinders"
 [3] "disp Displacement (cu.in.)"      "hp   Gross horsepower"
 [5] "drat Rear axle ratio"            "wt   Weight (1000 lbs)"
 [7] "qsec 1/4 mile time"              "vs   Engine(0=V,1=In-line)"
 [9] "am   Transmission(0=auto,1=man)" "gear Num forward gears"
[11] "carb Num carburetors"
```

We apply two **Approaches** to answering the questions:

(A) **T-test:** Analysis of the difference in means of `mpg` for the two transmission types
(B) **Regression:** Investigation of the impact of other variables in adjusting or replacing the transmission effect.

We have 5 **Conclusions** based on these analyses:

1. The two groups come from populations with statistically different means.
2. If the data is representative, mean MPG of *manual* transmission cars is approximately **7.2 MPG** higher than *automatic*, with a one-sided confidence interval of 3.9 MPG to infinity.
3. The adjustments of the transmission-type effect that are most helpful in prediction of `mpg` are `hp` and `wt`, both negative, and `vs`, which adds a positive adjustment for in-line engines as compared to V-type.
4. The *transmission* effect, with *horsepower*, *weight*, and **engine-type** held fixed, is approximately **2.4 MPG**, with manual again having higher `mpg` than automatic. These other engineered effects are the sources of more of the difference in `mpg` (**4.8 MPG** worth) than transmission type alone.
5. Due to low significance of the `am` coefficient (**Std. Error** of X), we also investigate other models not constrained to include `am`. The best of these, selected via cross-validation, was $E[mpg] = 39.6 - 0.49 \cdot carb - 1.29 \cdot cyl - 3.16 \cdot wt$.

2 Exploratory Data Analysis

Note that for our analyses to be meaningful, this sample of 32 cars must be representative of a larger, defined population of vehicles.

```
 [1] "Mazda RX4"           "Mazda RX4 Wag"       "Datsun 710"
 [4] "Hornet 4 Drive"      "Hornet Sportabout"   "Valiant"
 [7] "Duster 360"          "Merc 240D"           "Merc 230"
```

```
[10] "Merc 280"            "Merc 280C"           "Merc 450SE"
[13] "Merc 450SL"          "Merc 450SLC"         "Cadillac Fleetwood"
[16] "Lincoln Continental" "Chrysler Imperial"   "Fiat 128"
[19] "Honda Civic"         "Toyota Corolla"      "Toyota Corona"
[22] "Dodge Challenger"    "AMC Javelin"         "Camaro Z28"
[25] "Pontiac Firebird"    "Fiat X1-9"           "Porsche 914-2"
[28] "Lotus Europa"        "Ford Pantera L"      "Ferrari Dino"
[31] "Maserati Bora"       "Volvo 142E"
```

Note that the sample contains multiple Mercedes (7) and Mazdas (2). It may enhance prediction to consider a "Mercedes" and/or "Mazda" effect in our modeling. However, in **Assumption (b)** below, we restrict consideration to differences in fundamental engineering variables, implicitly assuming that such differences explain any brand specific effects.

In **Figure 1**, we examine integer predictors to decide whether to treat them as factors. We find value in treating `cyl` and `carb` as continuous: they show clear trends vs. other variables. And from a pairs plot, **Figure 2**, we see many strong correlations, so model selection should consider variance inflation.

3 Approach (A): T-test

Figure 3 shows that in this data, `mpg` varies with `am`. Mean mileage for manual is **7.2 mpg** higher than automatic. We compute p-value and confidence interval for the test of manual transmission *greater*.

```
t_test <- t.test(mpg~am,data=mtcars,alternative='less') #'less': level 2 is t.test base
# note: code for processing and formatting of output suppressed
```

```
p-value = 0.07%     95% confidence interval = 3.91 to Inf
```

Inference Given the p-value, and assuming rough normality, we are highly confident manual transmissions are associated with higher fuel economy in the populations from which these samples were drawn. Based on our sample values, the probability is only 1 in 20 that the interval above 3.9 MPG does not contain our true difference in means.

4 Approach (B): Regression

Approach **(B)** is under-specified. Having identified `mpg` and `am` as of interest, the "correct" choice of model still depends on selection of the appropriate subset of 9 other variables. This should be a function of variable/model significance, but also *Motor Trend's* interests. For significance testing, manual checking of p-values is in-viable: at least $2^9 = 512$ models with single predictors exist. Therefore, to fully specify and make the approach manageable, we must make additional assumptions.

Assumptions
In answering questions 1 ad 2, we assume *Motor Trend* values the following, in rough priority order:

(a) Model parsimony/simplicity
(b) Models with granular, causal variables that may clarify engineering trade-offs
(c) Predictiveness: good generalizability outside the training sample

4.1 Model Search

4.1.1 Test citations

This is a test citation: (Breiman and Spector 1992), (Rao, Fung, and Rosales 2008).

4.1.2 Model search discussion

1. Trade-off: Choose exact predictors during the inference process using R-squared for fixed number of predictors (do not detect over-fitting in predictor choice, only in number of predictors), or choose exact predictors during cross-validation (detect over-fitting in predictor choice, but potentially overfit the selection criterion)
2. LOOCV vs. K-fold CV vs. some point in between: a quagmire. Consensus is that positive bias on CV error is higher for K-fold, but no true consensus on the way variance changes on the spectrum of LOOCV to K-fold with small K.
3. Potential for over-fitting the selection criteria, especially for small sample size, leads to negative bias, but usage of low K for K-fold CV can have positive bias.
4. Use information theoretic result, or use cross-validation. Information theoretic result may be sufficient when precise variables chosen as part of inference on each fold, and there are indications that info theoretic results may be better than CV in such circumstances.

Due to **(a)**, we consider no interactions. From **(b)**, we exclude `qsec`, a summary metric. Due to **(c)**, we rank models using the **AIC** metric, which estimates model predictiveness outside the training sample. For OLS regression, the metric is $n \cdot Log(\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}) + 2k$, where $k = \#$ of parameters including estimate of residual error, an overfitting penalty. In fact, we use **AICc**, which corrects the penalty to be greater for small n by adding a term $\frac{2k(k+1)}{n-k-1}$ (note that this term varies based on model structure; this version only holds for Gaussian models). We use automated search to make evaluation of all 2^9 models feasible. Models are ranked from smallest to largest AICc. Only non-zero coefficients are shown, and only models with the variable of interest (`am`) are evaluated.

```r
if (!"MuMIn" %in% row.names(installed.packages())) {install.packages("MuMIn")}
library(MuMIn)
# preprocessing variables: remove unwanted perf. measure `qsec`; treat `gear` as factor
mtcars$qsec <- NULL; mtcars$gear <- as.factor(mtcars$gear)
globalmodel <- lm(mpg ~ ., data = mtcars, na.action = na.fail)

n_models <- 2^(length(model.frame(globalmodel)) - 1)

best_am_models <- dredge(globalmodel, subset = ~ am) # considers only models with `am`
best_am_models[1:10,]
```

Table 1: Best Models for MPG that Contain Transmission Type

	(Int)	am	carb	cyl	hp	vs	wt	df	logLik	AICc	delta	weight
322	34.0	2.08			-0.037		-2.9	5	-73.1	158.4	0.0	0.31
264	36.9	1.78	-0.75	-1.20			-2.5	6	-72.0	159.4	0.9	0.20
450	31.1	2.42			-0.030	1.8	-2.6	6	-72.0	159.4	1.0	0.19
326	36.1	1.48		-0.75	-0.025		-2.6	6	-72.1	159.6	1.2	0.17
262	39.4	0.18		-1.51			-3.1	5	-74.0	160.3	1.9	0.12

4.2 Model Inference & Interpretation of Coefficients

We investigate values of the `am` coefficient for the top models. Note the first 3 all round to 2, suggesting this is a good rough estimate of the adjusted transmission effect. Although our top model is only one AICc point lower than the next best model (model averaging is suggested via the weights, for differences less than 2), we focus attention on it, in the spirit of Assumption **(A)**.

Table 2: Coefficients of Best Model Including Transmission Type

	Estimate	Std. Error
(Intercept)	31.079	3.393
am	2.417	1.379
hp	-0.030	0.011
vs	1.786	1.327
wt	-2.591	0.917

The R^2 of this top model is 85%: it explains a high degree of sample variance with only 3 covariates. The above table contains no p-values, as after a search of 2^9 models, these would be inflated: since the procedure only selects ‘good’ models for consideration, we need to control the “False Discovery Rate”, which is the fraction of **all rejected** null hypotheses which are false (i.e., $(False\ Positives)/(False\ Positives{+}True\ Positives)$), not just α, which is the fraction of **all truly 0** results that are rejected (i.e., $(False\ Positives)/(False\ Positives+True\ Negatives)$). However, standard errors are provided. Increased engine **HP** accounts for a decrease of 3.0 MPG per hundred HP, increased **weight** accounts for a decrease of 2.6 MPG per thousand lbs, and **engine-type** accounts for a difference of 1.8 MPG, with inline engines having higher MPG than V-type engines. Note that significance of the the `am` coefficient, 2.4, is low. The *Estimate* over the *Std. Error*, or t-stat, is only 1.8. Two is near the $\alpha = 5\%$ threshold. Given this, we investigate models that do **not** include `am`. Here are the top 10 overall:

```r
# given a model, calculates mean-squared test error for leave-k-out cross validation
lkocv <- function(lm_model, folds, mpg){
        n_folds <- length(folds)
        modframe <- model.frame(lm_model)
        n <- names(modframe)
        sse <- numeric(n_folds)
        if (length(n) > 1){
                f <- as.formula(paste("mpg ~", paste(n[!n %in% "mpg"],
                                      collapse = " + ")))
                for (i in 1:n_folds) {
                        # iterate over all the folds, i; for each i...
                        # create model based on dataset leaving out fold i
                        trnd_model <- lm(formula = f, data =
                                modframe[ -folds[[i]], ])
                        # calculate and store sum of squared errors for each fold i
                        sse[i] <- sum((modframe[folds[[i]], "mpg"] -
                                predict(trnd_model, newdata =
                                        modframe[folds[[i]],]))^2)
                }
                } else {
                f <- as.formula("mpg ~ 1")
                for (i in 1:n_folds) {
                        # iterate over all the folds, i; for each i...
                        # create model based on dataset leaving out fold i
                        trnd_model <- lm(formula = f, data =
                                data.frame(mpg = mpg[ -folds[[i]] ]))
                        # calculate and store sum of squared errors for each fold i
                        sse[i] <- sum((mpg[ folds[[i]] ] -
                                predict(trnd_model, newdata =
                                        data.frame(mpg = mpg[ folds[[i]] ])))^2)
                }
        }
        # sum up the squared errors across all folds, divide by n, get MSE
        mse_nobs <- sum(sse)/nrow(modframe)
        mse_nobs
}


# function does n_iter iterations of cross-validation per fold, averages the results,
# returning out-of-sample r-squared
lkocv_iter <- function(lm_model, k.= k, n_iter.=n_iter, rseed. = rseed,
                       n_models. = n_models)
{
        set.seed(rseed.)
        modframe <- model.frame(lm_model)
        mpg <- modframe$mpg
        dev_mean <- mpg - mean(mpg)
        mse_iter <- numeric(n_iter.)
        n_sample <- nobs(lm_model)
        if (k. > 1) {
                for (i in 1:n_iter.) {
                        folds <- split(sample(n_sample), gl(ceiling(n_sample/k.),
                                                            k., n_sample))
                        # returns mean-squared-error of all observations in iteration i
                        mse_iter[i] <- lkocv(lm_model = lm_model, folds = folds, mpg)
                }

        } else {
                # leave-one-out cross-validation (loocv)
                h <- lm.influence(lm_model)$hat
                mse_iter <- (residuals(lm_model)/(1 - h))^2
        }
        if (.GlobalEnv$n %% 10 == 0) cat(sep = "", "\n")
        assign("n", .GlobalEnv$n + 1, .GlobalEnv)
        if (.GlobalEnv$n <= n_models.) {
                cat(sep = "", "[", .GlobalEnv$n,"/", n_models.,"]")
        }
        # 1 - out-of-sample R-squared
        mean(mse_iter)/mean(dev_mean^2)
}
```

4.2.1 Model Investigation: Leave K-out Cross Validation

4.2.1.1 Eight-Fold Cross Validation

```
assign(x = "n", value = 0, envir = .GlobalEnv) # for running as RStudio NB/console
n_iter <- 25
rseed <- 19720921
k <- 4
model_sel_leave4_out_df <- dredge(globalmodel, rank = lkocv_iter)
prnt_model_sel_table(model_sel_leave4_out_df, metric_name = "1-Rsq_l4ocv",
                  caption = "Best Overall Models for MPG (L4OCV)")
```

4.2.1.2 Leave One Out Cross Validation

```r
assign(x = "n", value = 0, envir = .GlobalEnv)
metric_name <- "1-Rsq_loocv"
caption <- "Best Overall Models for MPG (LOOCV)"
rseed <- 19720921
k <- 1
n_iter <- 1
model_sel_leave1_out_df <- dredge(globalmodel, rank = lkocv_iter)
```

```r
prnt_model_sel_table(model_sel_leave1_out_df, metric_name = "1-Rsq_loocv",
                     caption = caption)
```

Table 3: Best Overall Models for MPG (LOOCV)

	(Int)	am	carb	cyl	disp	drat	gear	hp	vs	wt	df	1-Rsq_loocv	delta
263	39.6		-0.49	-1.29						-3.16	5	0.2028	0.0000
325	38.8			-0.94				-0.018		-3.17	5	0.2043	0.0015
450	31.1	2.42						-0.030	1.79	-2.59	6	0.2066	0.0037
391	39.9		-0.50	-1.34					-0.19	-3.14	6	0.2072	0.0044
333	40.8			-1.29	0.012			-0.021		-3.85	6	0.2085	0.0057
271	40.0		-0.47	-1.38	0.002					-3.29	6	0.2090	0.0062
453	38.5			-0.91				-0.018	0.15	-3.18	6	0.2095	0.0067
261	39.7			-1.51						-3.19	4	0.2096	0.0068
326	36.1	1.48		-0.75				-0.025		-2.61	6	0.2103	0.0074
264	36.9	1.78	-0.75	-1.20						-2.48	6	0.2120	0.0092

4.2.1.3 Four-Fold Cross Validation

```r
assign(x = "n", value = 0, envir = .GlobalEnv) # for running as RStudio NB/console
n_iter <- 50
rseed <- 20101008
k <- 8
model_sel_leave8_out_df <- dredge(globalmodel, rank = lkocv_iter)
```

```r
prnt_model_sel_table(model_sel_leave8_out_df, metric_name = "1-Rsq_l8ocv",
                     caption = "Best Overall Models for MPG (L8OCV)")
```

Table 4: Best Overall Models for MPG (L8OCV)

	(Int)	am	carb	cyl	disp	drat	gear	hp	vs	wt	df	1-Rsq_l8ocv	delta
263	39.6		-0.49	-1.29						-3.16	5	0.2130	0.0000
261	39.7			-1.51						-3.19	4	0.2171	0.0041
325	38.8			-0.94				-0.018		-3.17	5	0.2188	0.0058
391	39.9		-0.50	-1.34					-0.19	-3.14	6	0.2190	0.0060
271	40.0		-0.47	-1.38	0.002					-3.29	6	0.2211	0.0082
389	38.7			-1.36					0.52	-3.25	5	0.2230	0.0100
279	33.3		-0.65	-1.08		1.251687				-2.86	6	0.2231	0.0101
322	34.0	2.08						-0.037		-2.88	5	0.2234	0.0104
450	31.1	2.42						-0.030	1.79	-2.59	6	0.2238	0.0109
333	40.8			-1.29	0.012			-0.021		-3.85	6	0.2241	0.0111

We see above that models involving `am` do not appear until rank 4 and below. But, given **(A)**, it is prudent to add the top model overall, involving `carb`, `cyl`, and `wt`, to the results presented to *Motor Trend*. Compared to the model containing `am`, it has one fewer parameter - therefore less likely to fall victim to overfitting, and only slightly lower R^2, equal to 84%. Also, the ratios of *Std. Errors* to *Estimates* make all coefficients appear significant. This model implies, though, that none of the variance in MPG is really due to transmission type, but to the combined effect of # of carburetors, cylinders, and weight.

4.3 Model Diagnostics

We run base R's standard plots in **Figure 4**. Though the smoother line in *Residuals vs Fitted* shows curvature, pointing to possible quadratic terms, the trend is not pronounced except for the 3 labeled points (*Toyota Corolla, Fiat 128,* and *Chrysler Imperial*). These have notably higher MPG than the trend. *Normal Q-Q* shows a somewhat right skewed distribution beyond 1 normal quantile. But the deviations are not extreme, except for the 3 labeled points, and the lowest. The lowest, given by the code below, is *Mazda RX4*.

```
best_am_model <- lm(mpg~am+hp+vs+wt,mtcars)
row.names(mtcars)[which.min(best_am_model$residuals)]
```

Scale-Location shows some heteroskedasticity. In *Residuals vs Leverage*, all points are inside 0.5 *Cook's distance*, indicating stable βs. Finally, from package `car`, we use `vif()` to calculate the Variance Inflation Factors and evaluate collinearity. All are under 5, causing no alarm (code suppressed to conserve space). VIFs:

Table 5: VIFs for Best Model Including Transmission Type

am	hp	vs	wt
2.35	2.79	2.22	3.99

Given, especially, the heteroskedasticity, we examine the diagnostics for the top model overall, too (**Figure 5**, `mpg ~ carb + cyl + wt`). It does not appear markedly better anywhere, and it appears slightly worse on the **Normal Q-Q** evaluation.

5 Figures

5.1 Treatment of Integer-Valued Variables

Plots of integer vs. continuous-valued variables provide guidance on whether to treat integer variables as continuous or factor. Variable `gear` behaves as a factor: continuous-valued functions do not vary in linear relationship.

```
data(mtcars)
int_plots_df <- data.frame(mpg=mtcars$mpg,hp=mtcars$hp,cyl=mtcars$cyl,carb=mtcars$carb,
                           gear=mtcars$gear, qsec=mtcars$qsec)
if (!"tidyr" %in% rownames(installed.packages())) install.packages("tidyr")
library(tidyr)
int_plots_df <- gather(data = int_plots_df, key = x_var, value = x, -mpg, -hp, -qsec)
int_plots_df <- gather(data = int_plots_df, key = y_var, value = y, -x_var, -x)
if (!"ggplot2" %in% rownames(installed.packages())) install.packages("ggplot2")
library(ggplot2)
g_integer_vars <- ggplot(int_plots_df, aes(x=x,y=y)) +
                  facet_grid(y_var ~ x_var, scales = "free") +
                  geom_point() + geom_smooth(method = "lm") +
labs(title = "Variable 'gear' can be treated as a factor, 'carb' & 'cyl' as continous")
print(g_integer_vars)
```

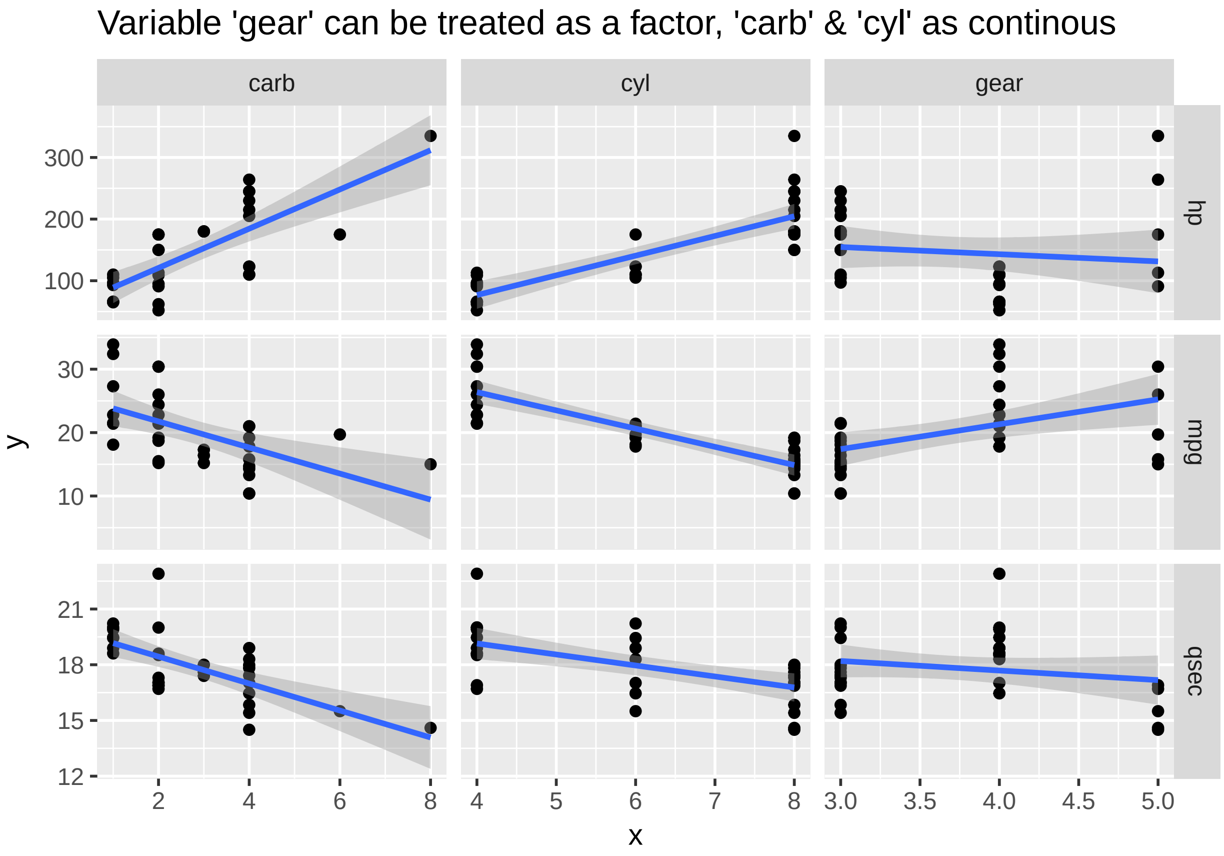

Figure 1: Plot of continuous vs. factor variables in mtcars data

5.2 Pairs plot for mtcars dataset

The pairs plot shows many strong correlations.

```
if (!"GGally" %in% rownames(installed.packages())) {install.packages("GGally")}
library(GGally)
g_pairs <- ggpairs(mtcars, mapping=aes(color=am, alpha = 0.7),
           lower = list(continuous = wrap(ggally_smooth, size = 1)),
           diag = list(continuous = "barDiag"), upper = list(continuous =
           wrap(ggally_cor,size=2, mapping=aes(color=am,alpha=1))), axisLabels = 'none')
print(g_pairs)
```

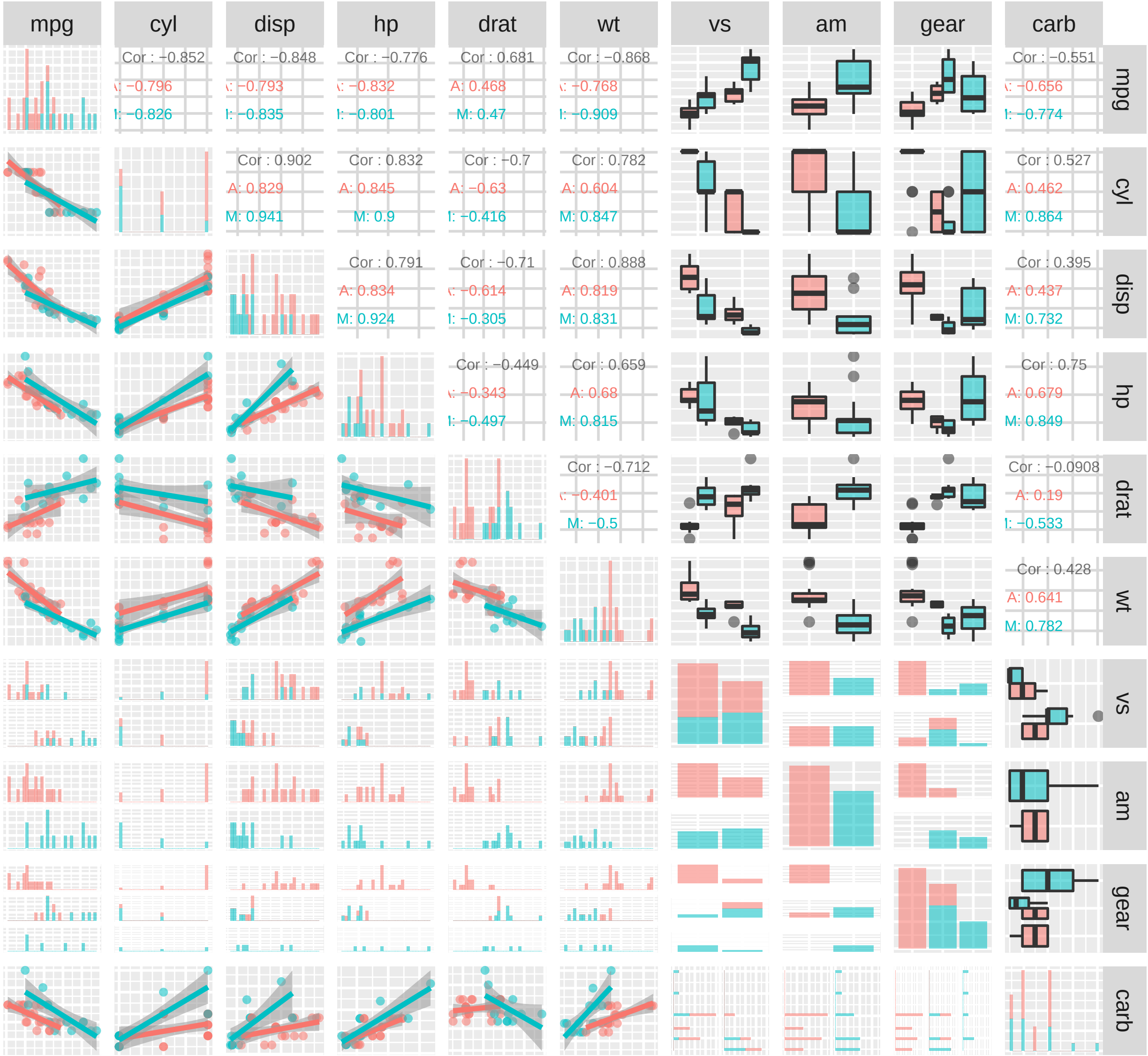

Figure 2: Pairs Plot of Motor Trend Cars Dataset

5.3 Violin Plot for Manual vs. Automatic Transmissions

The violin plot depicts association between transmission type and `mpg`.

```
g_violin <- ggplot(mtcars, aes(x = am, y = mpg)) +
            geom_violin() +
            # geom_dotplot(binaxis='y', stackdir='center', dotsize=1) +
            stat_summary(fun.y=mean, geom="point", shape=23, size=4, aes(fill = am)) +
            scale_fill_discrete(name="Mean MPG")+
            labs(title = "Average MPG for manual transmissions is significantly higher")
print(g_violin)
```

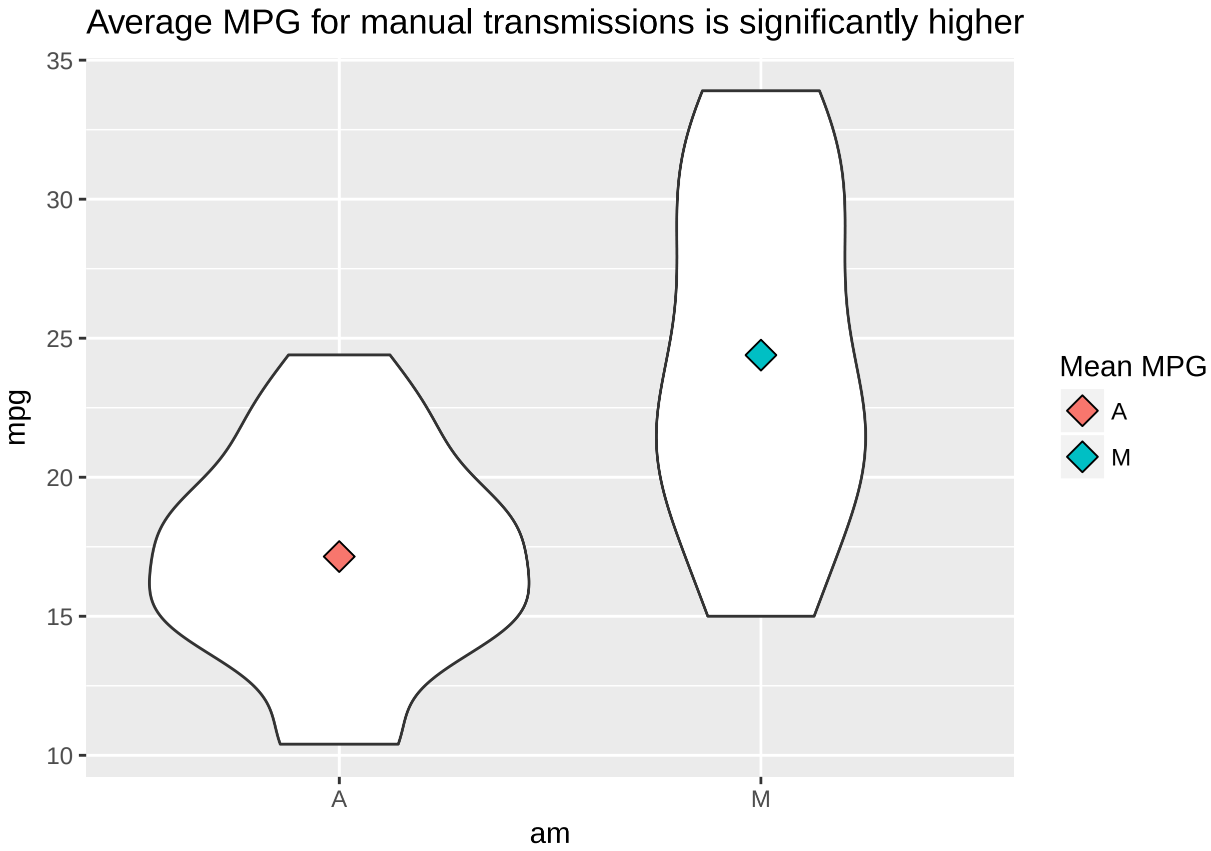

Figure 3: Violin plot of MPG vs. Transmission type

5.4 Pairs Plot for Variables of Best Constrained and Unconstrained Models

```
g_model_pairs <- ggpairs(mtcars[,c("mpg","hp","am","carb","cyl","vs","wt")],
                 mapping=aes(color=am, alpha = 0.7),
                 upper = list(continuous = wrap(ggally_smooth, size = 1)),
                 diag = list(continuous = "barDiag"), lower = list(continuous =
                 wrap(ggally_cor, size=3, mapping=aes(color=am,alpha=1))),
                 axisLabels = 'none')
print(g_model_pairs)
```

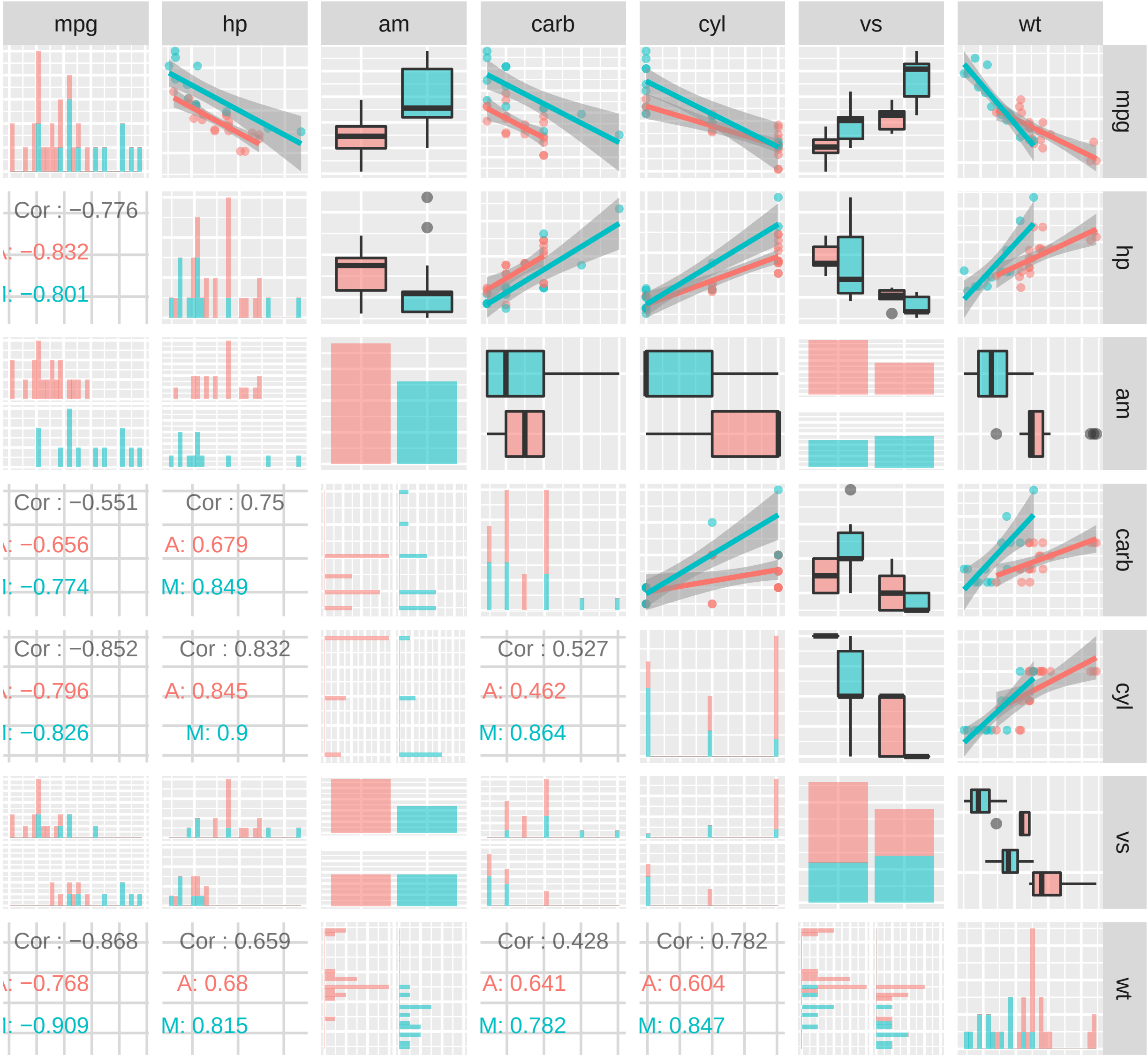

Figure 4: Pairs Plot of Variables of Best Constrained and Unconstrained Models

5.5 Model Diagnostics

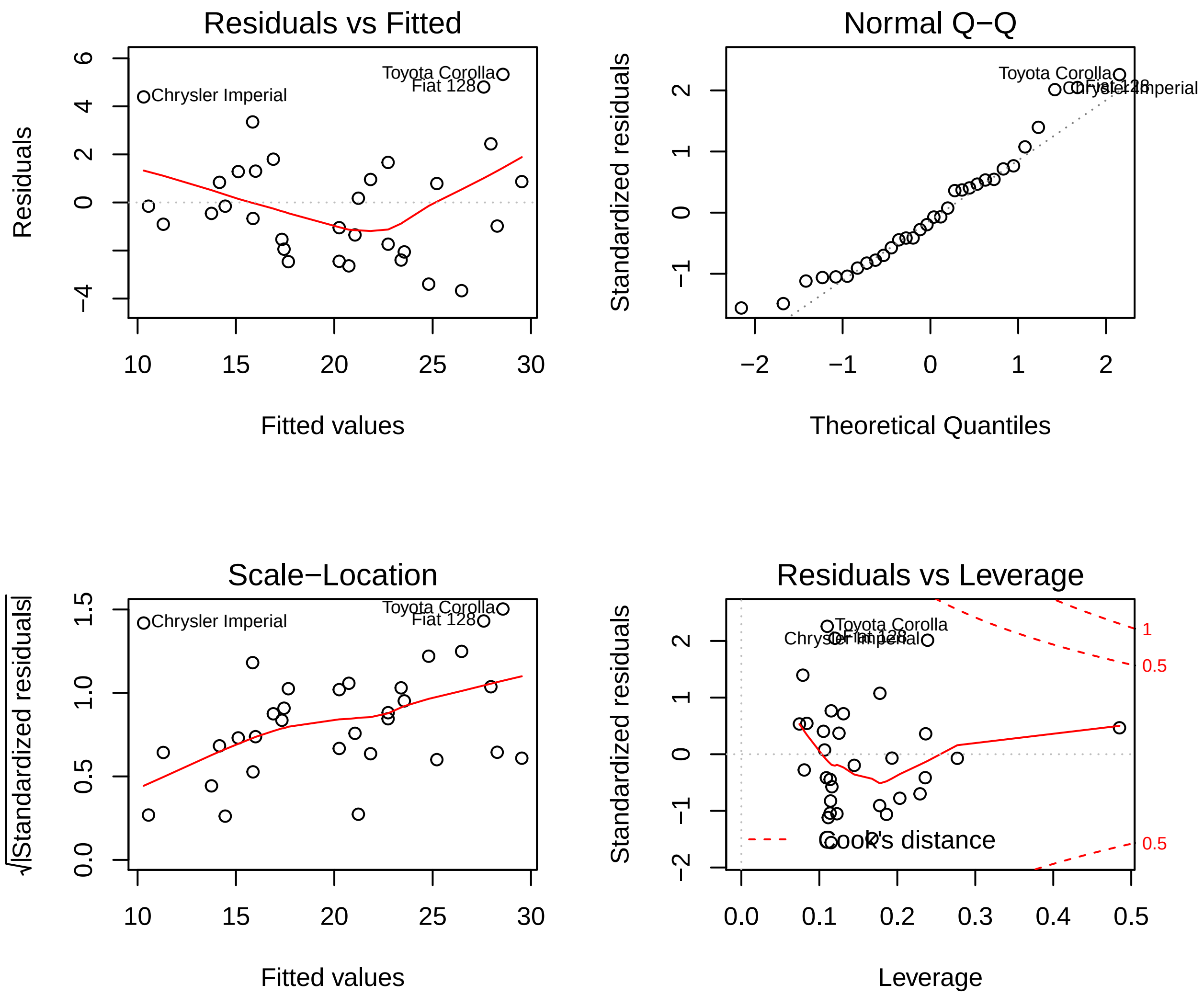

Figure 5: Diagnostic Plots - Best Model that Includes Transmission Type

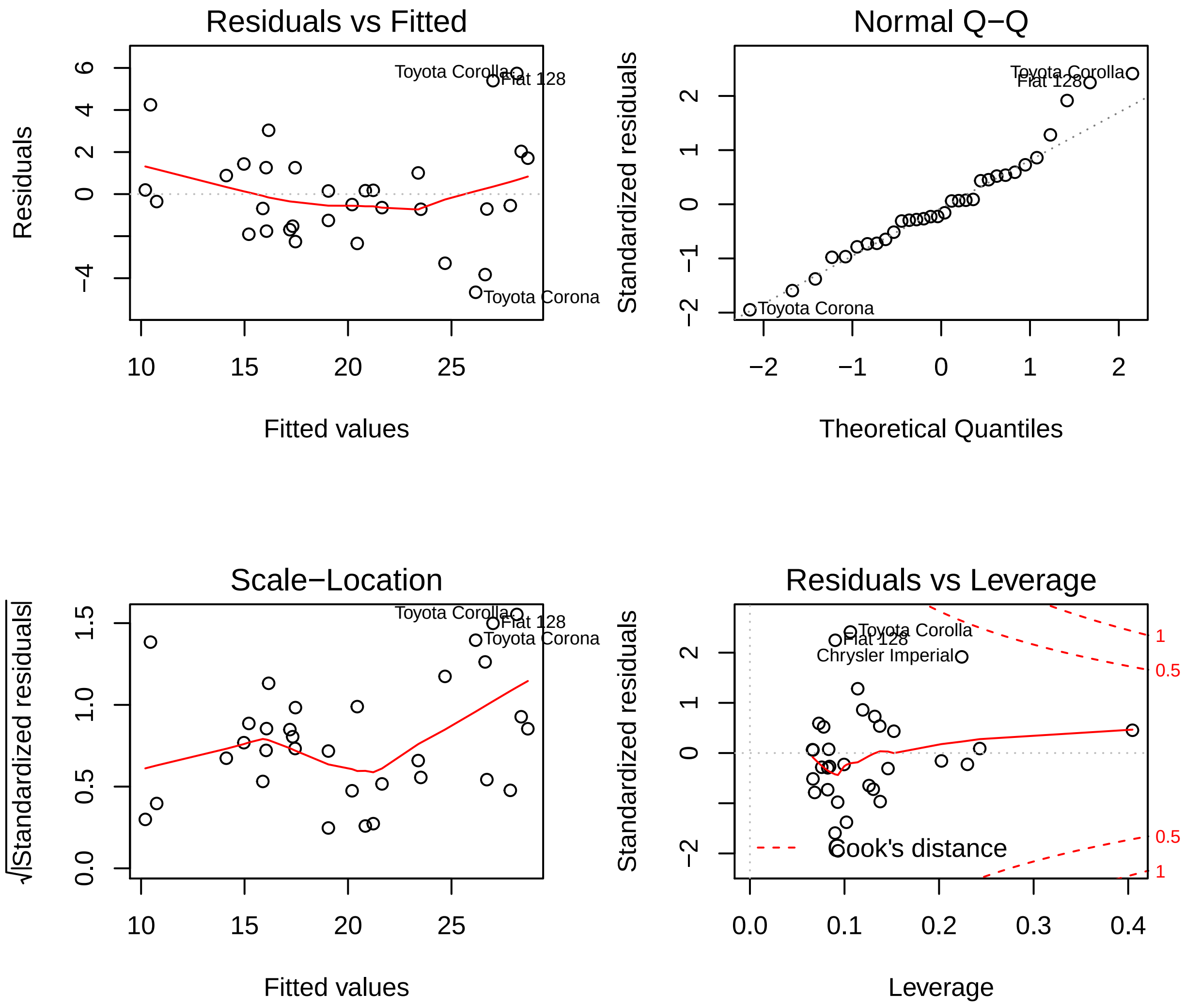

Figure 6: Diagnostic Plots - Best Overall (Unconstrained) Model

Bibliography

Breiman, Leo, and Philip Spector. 1992. “Submodel Selection and Evaluation in Regression: The X-Random Case.” *International Statistical Review / Revue Internationale de Statistique* 60 (3). [Wiley, International Statistical Institute (ISI)]: 291–319. http://www.jstor.org/stable/1403680.

Rao, R. Bharat, Glenn Fung, and Romer Rosales. 2008. “On the Dangers of Cross-Validation: An Experimental Evaluation.” In *Proceedings of the 2008 Siam International Conference on Data Mining*, 588–96. doi:10.1137/1.9781611972788.54.