Network analysis project

City street analysis from a network and topology perspective

Riku Laine

26 helmikuu, 2019

Introduction

The purpose of this document is to illustrate the statistical model fitting procedure used to infer street network properties that affect travel times in cities.

The data

- Street network metrics from 197 cities. 36 different variables.
- Cities from Inrix scorecard listing.
- It was observed that some of the areas obtained were close to zero square kilometers or clearly too small to be correctly mapped (see Table 1). It was decided that if the logarithm of the graph area is bigger than the mean of the logarithms of the areas subtracted by one standard deviation of the log areas, the city would be included in the analysis. I.e. include the city if `log(graph_area) > mean(log(graph_area)) - sd(log(graph_area))`. Subsequently 17 cities were removed.

```
# Import data
metrics <- read.csv2("C:/Users/Riku_L/network-analysis-uoh/data/merged_traffic_network_statistics.csv",
                     header = T, stringsAsFactors = F, dec=".")

# Remove columns with all missing values
metrics <- metrics[, apply(metrics, 2, function(x) {!all(is.na(x))})]

# Remove columns containing ID numbers and other redundant columns.
metrics <- metrics[, !(colnames(metrics) %in%
                         c("city_name.y", "pos", "hours_congestion", "year_change", "cost",
                           "pagerank_max_node", "pagerank_min_node"))]

# Print 20 smallest graph areas
smallest <- head(order(metrics$graph_area), 20)

kable(metrics[smallest, c('city_name', 'graph_area')],
      digits = 0,
      caption = "20 smallest graph areas (in m^2^).",
      row.names = F)
```

Table 1: 20 smallest graph areas (in m^2).

city_name	graph_area
gold coast	6425
halifax, ns	336663
copenhagen	1563934
johannesburg	2475008
riyadh	2623295

city_name	graph_area
perth	5737402
quito	5795797
brisbane	8673674
edinburgh	9323960
durban	12132292
adelaide	14744114
geneva	18968697
brussels	27433896
la paz	38598092
bilbao	41308764
glasgow	46921762
lille	49505449
melbourne	51593188
southampton	56571368
leicester	59221795

```
# Remove observations according to prespecified rule
log_area <- log(metrics$graph_area)

cutpoint <- mean(log_area) - sd(log_area)

cat("Cutpoint in sq. kms:", exp(cutpoint)/1e6)
```

```
## Cutpoint in sq. kms: 49.80604
```

```
metrics <- subset(metrics, log_area > cutpoint)

row.names(metrics) <- NULL # Reset rownames
```

Analysis

The goal is to build a statistical model to explain the variable `inner_mile`: "The time it takes to travel one mile into the central business district during peak hours" (Inrix).

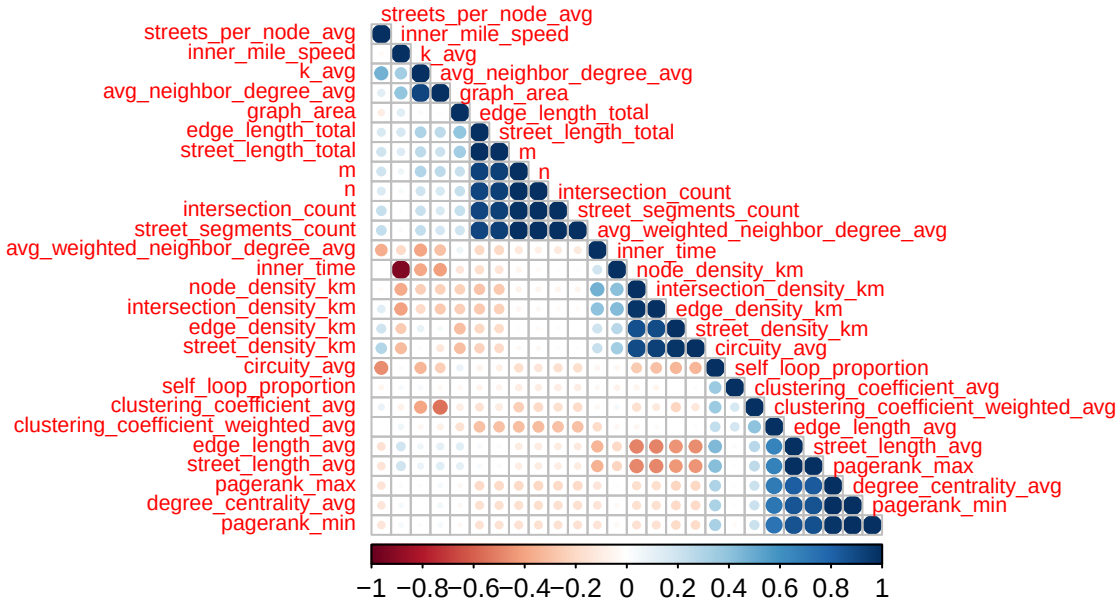
Correlations

It as expected that the variables show extreme values of correlation. Below are figures for Pearson and Spearman rank correlation. Latter indicates correlations with the variable ranks, order of greatness, and doesnt expect linear dependence. Illuminating example given in the relationship with the maximum PageRank value `pagerank_max` and number of nodes `n`.

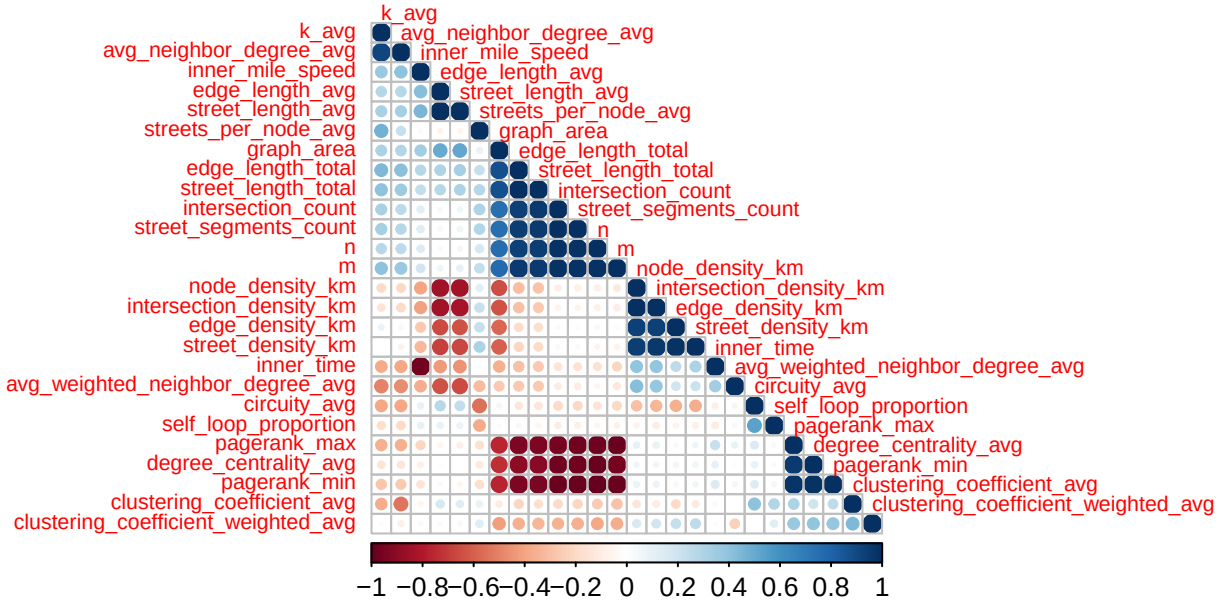
```
# Correlation plots
corrplot(cor(metrics[,-1]), order = "hclust", tl.cex = 0.7,
         method = "circle", type = "lower", tl.srt = 0.1)

title(main = "Correlation plots, ordering by hierarchical clustering",
      sub = "Pearson correlation")

corrplot(cor(metrics[,-1], method = "spearman"), order = "hclust",
         tl.cex = 0.7,  method = "circle", type = "lower", tl.srt = 0.1)
```

```r
title(sub = "Spearman rank correlation")

par(mfrow=c(1,2))

plot(metrics$n, metrics$pagerank_max, main = "Linear scale")
plot(metrics$n, metrics$pagerank_max, log = 'xy', main = "Log-log scale")

par(mfrow=c(1,1))
```

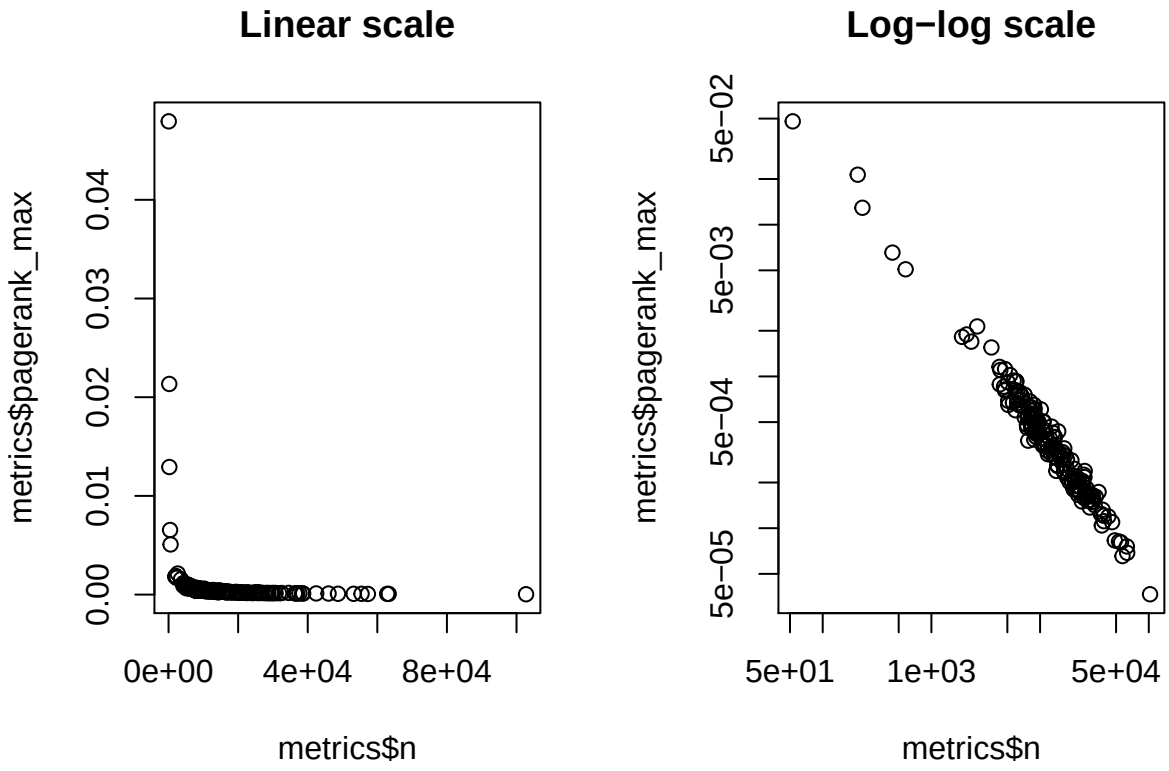
Correlation plots, ordering by hierarchical clustering



Pearson correlation



Spearman rank correlation



Modelling & Inference

It is hypothesized that the dependence between the properties and traffic time is linear.

Variable selection

Basis for variable selection is reported at this document. Some of the variable selection and construction was based on the assumption that if a street was a two-way street the library broke it down as two one-directional edges between nodes, i.e. initersections.

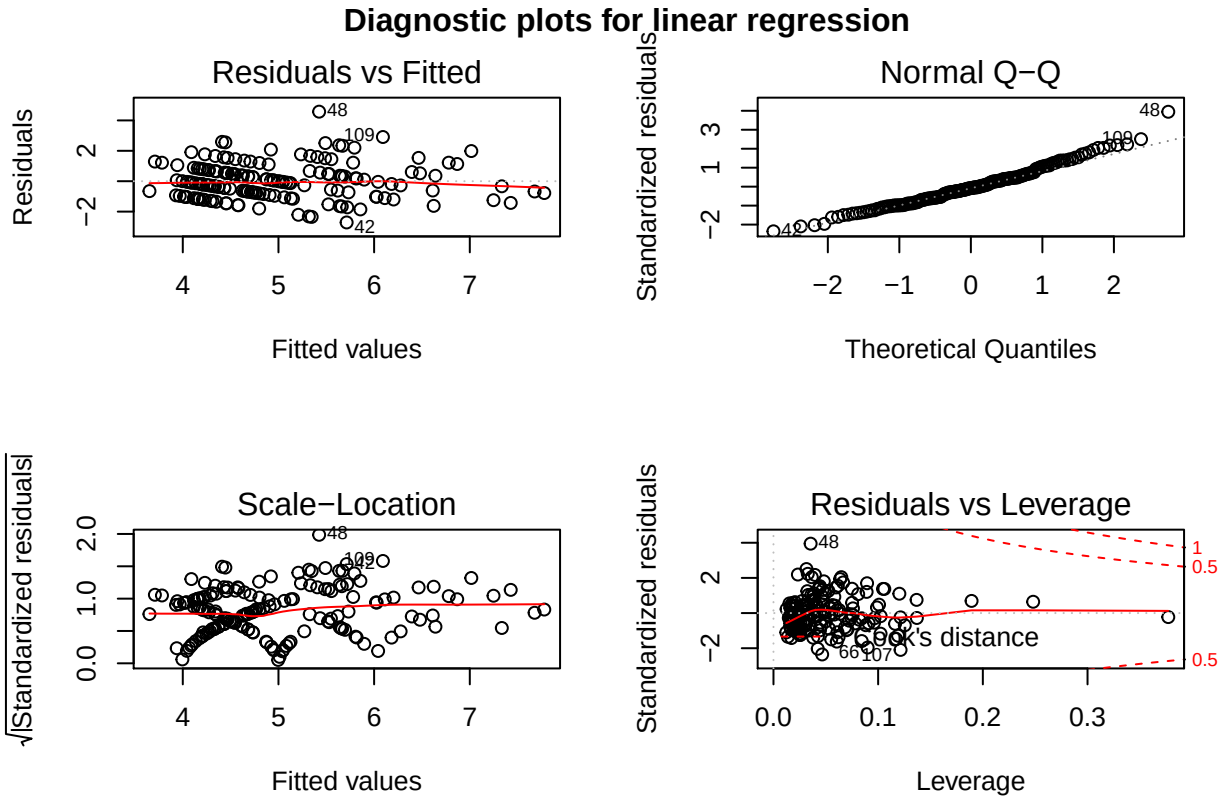
During the model fitting seven misspecified networks were identified by Cook's distance from Acapulco, Waterloo (Ontario), Villahermosa, Mecca, Leicester, Belfast and Medellin. For all of them but Waterloo, Ontario, the misspecification was the result of OpenStreetMap service lacking their respective border polygon. Usually the service had the location of the cities as a point, but lacked the polygon, and then the next polygon in list (obtainable via search from here) was usually in the US. Waterloo, Ontario was excluded from analysis as it was evident that the Inrix analysis also included the street network of the neighbouring city of Kitchener.

```
# Construct additional variable
metrics$percentage_twoway <- (metrics$m - metrics$street_segments_count) / metrics$street_segments_coun

regression_vars <- c("inner_time", "k_avg", "edge_length_avg", "intersection_density_km",
                     "street_density_km", "self_loop_proportion", "degree_centrality_avg",
                     "clustering_coefficient_avg", "percentage_twoway")
```

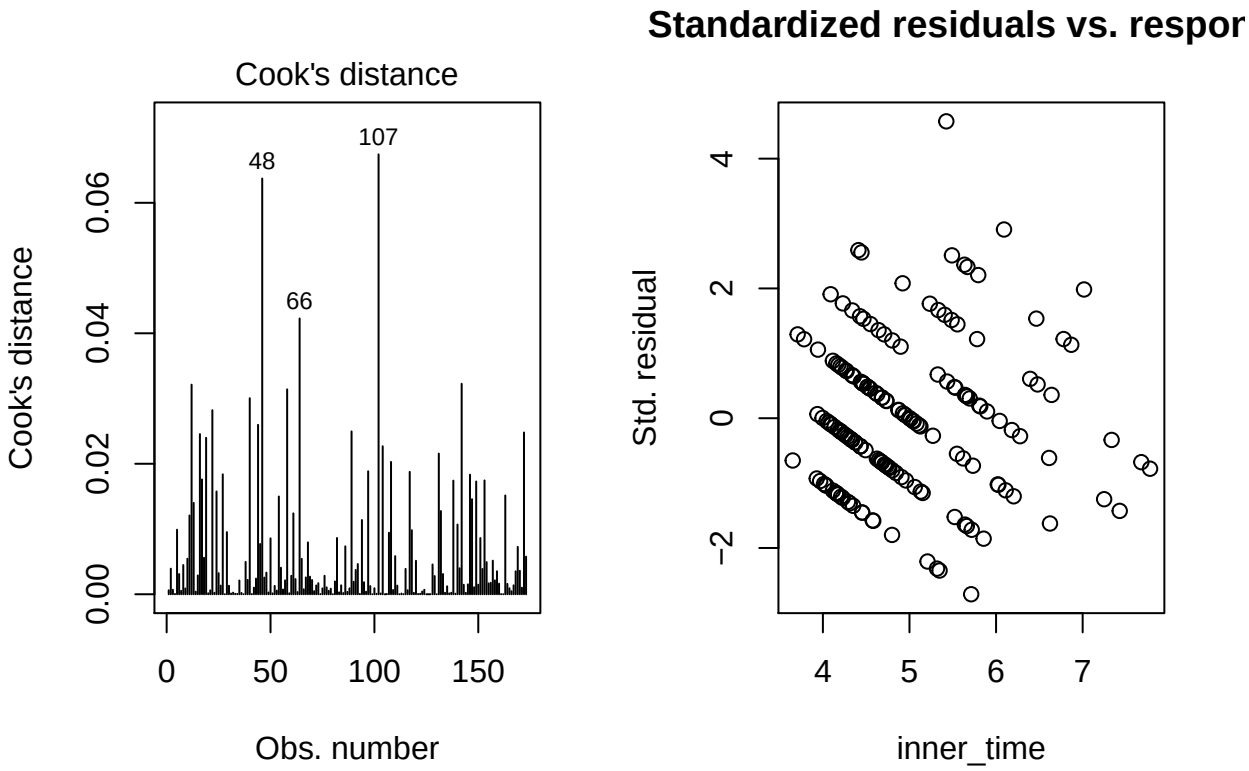
```r
# Define outliers to omit
outliers_to_omit <- -c(1,168,173, 92, 77, 13, 93)

# Fit linear model
met_linear <- lm(inner_time ~ ., data = metrics[outliers_to_omit, regression_vars])

# Print model summaries
summary(met_linear)
```

```
##
## Call:
## lm(formula = inner_time ~ ., data = metrics[outliers_to_omit,
##     regression_vars])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7127 -0.7785 -0.0866  0.6061  4.5748
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  8.596e+00  1.449e+00   5.931 1.74e-08 ***
## k_avg                       -4.115e-01  2.805e-01  -1.467  0.14430
## edge_length_avg              1.169e-03  4.026e-03   0.290  0.77189
## intersection_density_km      4.094e-02  1.726e-02   2.371  0.01889 *
## street_density_km           -1.096e-04  1.044e-04  -1.050  0.29542
## self_loop_proportion        -5.318e+00  2.682e+01  -0.198  0.84309
## degree_centrality_avg       -3.721e+01  3.409e+02  -0.109  0.91321
## clustering_coefficient_avg -9.511e+00  7.422e+00  -1.281  0.20185
## percentage_twoway           -2.691e-02  9.285e-03  -2.898  0.00427 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.18 on 164 degrees of freedom
## Multiple R-squared:  0.3546, Adjusted R-squared:  0.3231
## F-statistic: 11.26 on 8 and 164 DF,  p-value: 1.195e-12
```

```r
# Plot diagnostic plots
par(mfrow=c(2,2))
plot(met_linear)
title("Diagnostic plots for linear regression", outer = T, line = -2)
```




```
par(mfrow=c(1,2))
plot(met_linear, which = 4)
plot(fitted(met_linear), residuals(met_linear, "pearson"),
     ylab = "Std. residual", xlab = "inner_time", main = "Standardized residuals vs. response")
```



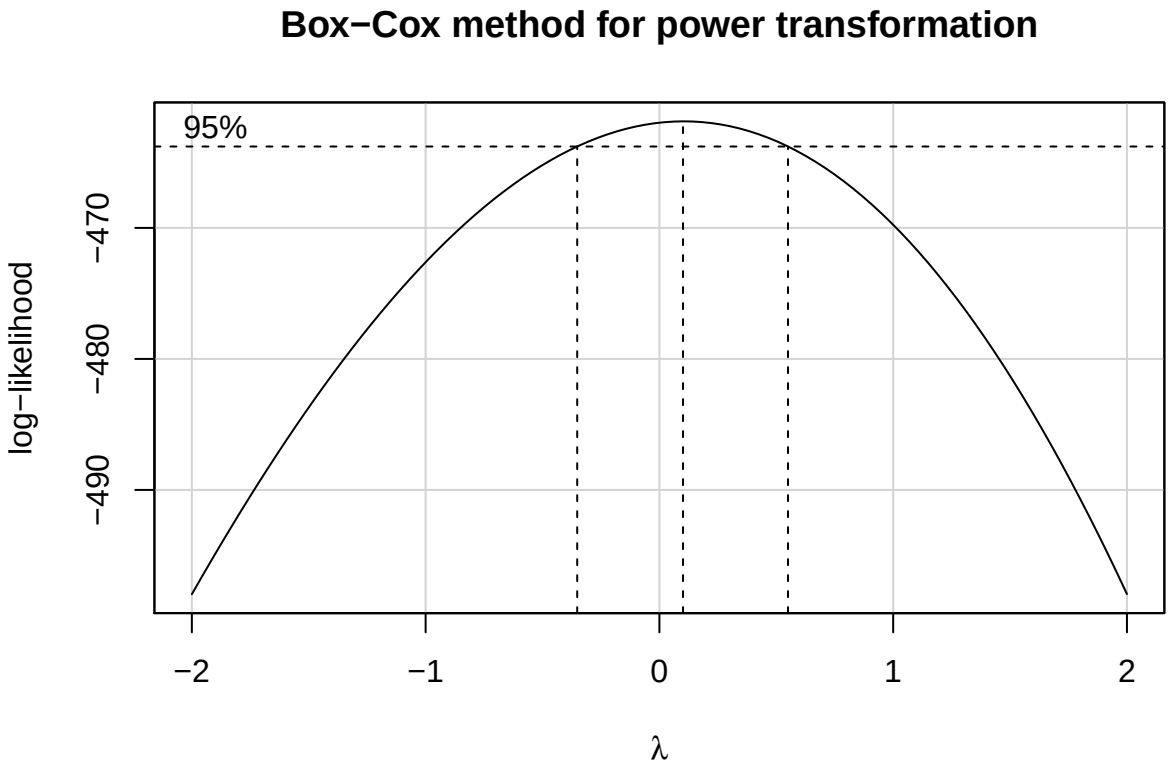

```
par(mfrow=c(1,1))

# Test for constant variance
ncvTest(met_linear)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 5.908333, Df = 1, p = 0.015069
```

QQ-plot indicates that that there are some deviations from the assumptions. `ncvTest` confirms that variance is not constant. From the Cook's distance figure it is evident that no significant outliers are anymore present. Observations 48, 66 and 107 (Dublin, Helsinki and Naberezhnye Chelny) have been validated to be correct.

Let's conduct Box-Cox to obtain best power transformation for our response:

```
boxCox(met_linear)
title("Box-Cox method for power transformation")
```




```
summary(powerTransform(met_linear))
```

```
## bcPower Transformation to Normality
##    Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## Y1    0.1048           0      -0.3456       0.5553
##
## Likelihood ratio test that transformation parameter is equal to 0
##  (log transformation)
##                             LRT df   pval
## LR test, lambda = (0) 0.2069113  1 0.6492
##
## Likelihood ratio test that no transformation is needed
##                            LRT df       pval
## LR test, lambda = (1) 15.7905  1 7.0757e-05
```

From the function summary it is seen that transformation is needed and that it should be log transformation. Let's redefine model with response transformed to log scale.

Notable is that now interpreting as in wiley etc etc.

```
# Fit linear model with response in log scale.
met_log <- lm(log(inner_time) ~ ., data = metrics[outliers_to_omit, regression_vars])

# Print model summaries
summary(met_log)
```

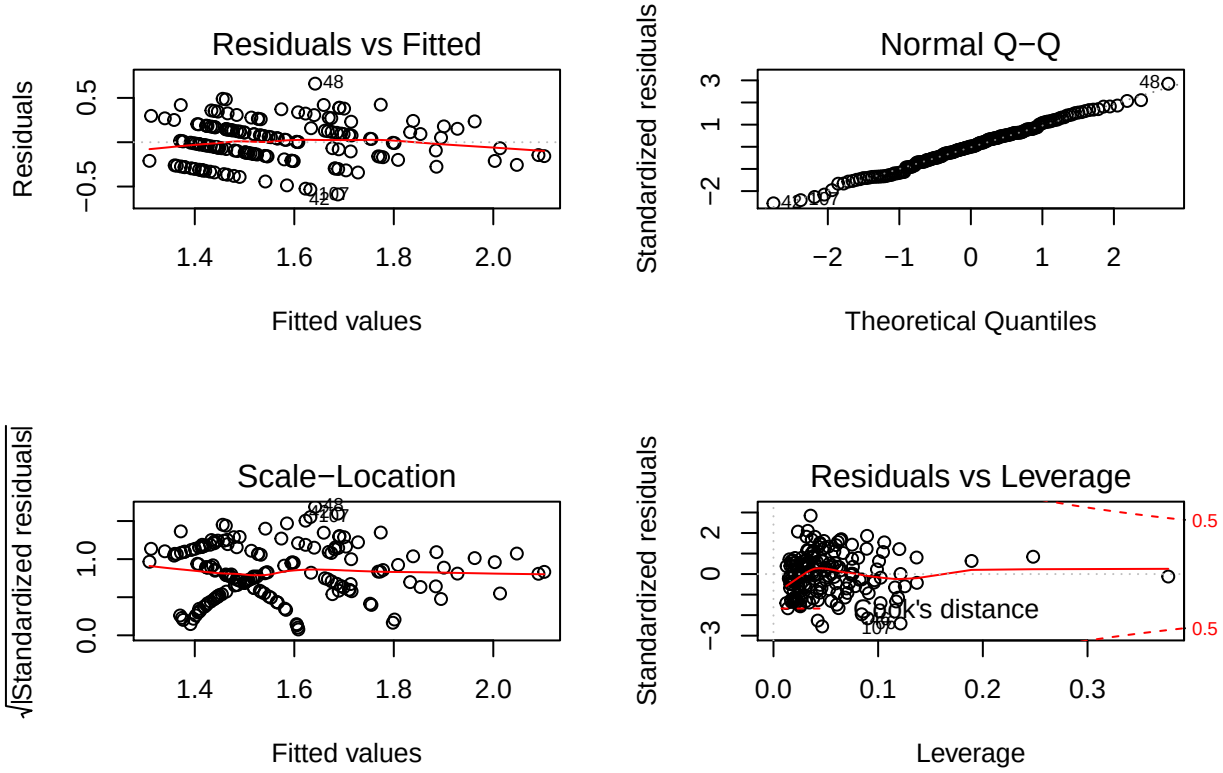
```
##
## Call:
```

```
## lm(formula = log(inner_time) ~ ., data = metrics[outliers_to_omit,
##     regression_vars])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.58909 -0.15655  0.00232  0.14652  0.66068
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  2.245e+00  2.905e-01   7.730 1.02e-12 ***
## k_avg                       -6.451e-02  5.621e-02  -1.148  0.25275
## edge_length_avg              1.578e-04  8.067e-04   0.196  0.84517
## intersection_density_km      7.514e-03  3.460e-03   2.172  0.03130 *
## street_density_km           -1.958e-05  2.092e-05  -0.936  0.35049
## self_loop_proportion         1.573e+00  5.375e+00   0.293  0.77022
## degree_centrality_avg        6.183e+00  6.831e+01   0.091  0.92799
## clustering_coefficient_avg -2.116e+00  1.487e+00  -1.423  0.15667
## percentage_twoway           -5.838e-03  1.861e-03  -3.138  0.00202 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2364 on 164 degrees of freedom
## Multiple R-squared:  0.3354, Adjusted R-squared:  0.3029
## F-statistic: 10.34 on 8 and 164 DF,  p-value: 1.134e-11
```

```
# Test for non-constant variance
ncvTest(met_log)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.001655283, Df = 1, p = 0.96755
```

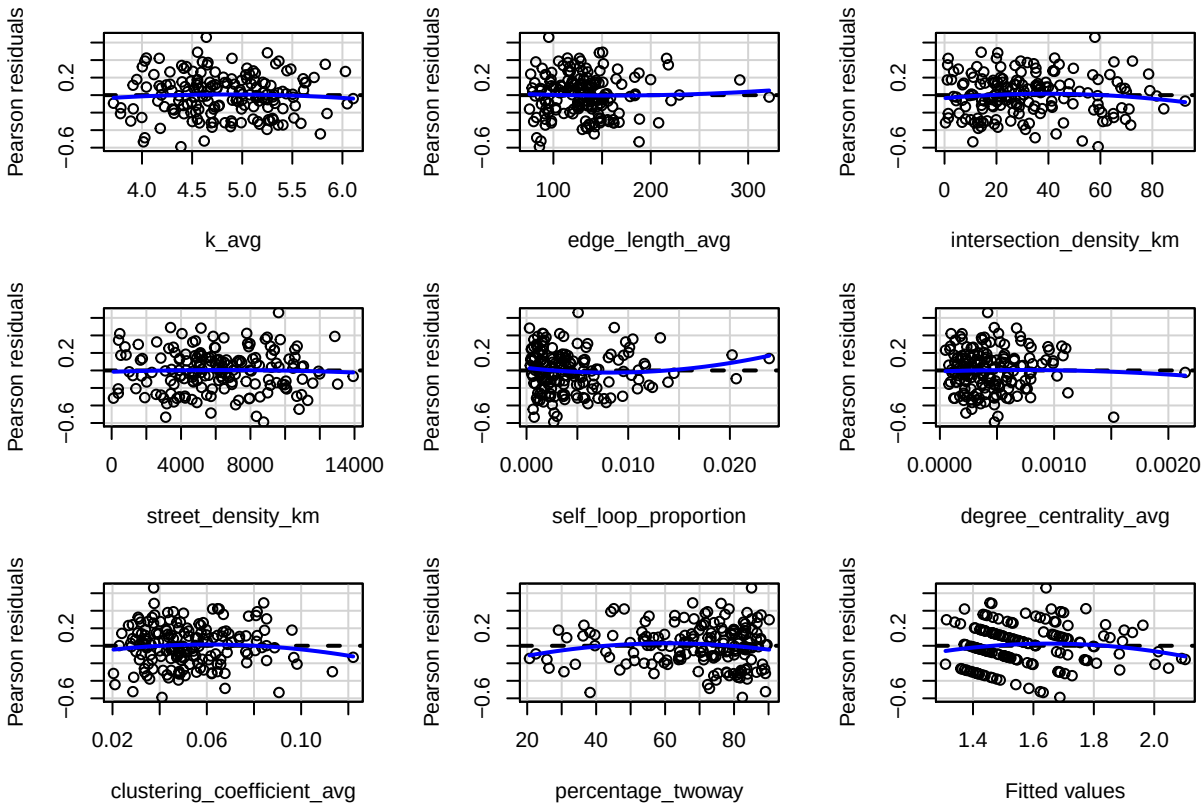
```
# Diagnostic plots
par(mfrow=c(2,2))
plot(met_log)
```




```
par(mfrow=c(1,1))

# Test for normally distributed residuals
shapiro.test(met_log$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  met_log$residuals
## W = 0.99484, p-value = 0.8121
```

```
# Residual vs predictor / fitted values plots
# and Tukey's test for nonadditivity
residualPlots(met_log)
```




```
##                              Test stat Pr(>|Test stat|)
## k_avg                          -0.6018           0.5482
## edge_length_avg                 0.5588           0.5771
## intersection_density_km        -1.4523           0.1483
## street_density_km              -0.3777           0.7061
## self_loop_proportion            1.3478           0.1796
## degree_centrality_avg          -0.3764           0.7071
## clustering_coefficient_avg     -0.9994           0.3191
## percentage_twoway              -1.5325           0.1273
## Tukey test                     -1.5229           0.1278
```

```
# Global Validation of Linear Models Assumptions, see ref.
gvlma(met_log)
```

```
##
## Call:
## lm(formula = log(inner_time) ~ ., data = metrics[outliers_to_omit,
##     regression_vars])
##
## Coefficients:
##                  (Intercept)                        k_avg
##                    2.245e+00                   -6.451e-02
##              edge_length_avg      intersection_density_km
##                    1.578e-04                    7.514e-03
##           street_density_km         self_loop_proportion
##                   -1.958e-05                    1.573e+00
##       degree_centrality_avg  clustering_coefficient_avg
```

```
##                  6.183e+00                  -2.116e+00
##          percentage_twoway
##                 -5.838e-03
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = met_log)
##
##                      Value p-value                Decision
## Global Stat        3.15909  0.5316 Assumptions acceptable.
## Skewness           0.01399  0.9059 Assumptions acceptable.
## Kurtosis           0.61045  0.4346 Assumptions acceptable.
## Link Function      2.42683  0.1193 Assumptions acceptable.
## Heteroscedasticity 0.10783  0.7426 Assumptions acceptable.
```

The unconstant variance has now been corrected as proposed by Weisberg. The tails of the QQ-plot do not deviate anymore and the residuals are normally distributed. Also Tukey's tests for nonadditivity returned non-significant.

Therefore it can be concluded that the model fits.

Results

```
summ <- summary(met_log)

coefs <- signif(100*(exp(summ$coefficients) - 1)[-1, 1], 3)
pvals <- round(summ$coefficients[-1, 4], 3)
signifs <- ifelse(pvals < 0.05, "Yes", "No")
tbl <- cbind(coefs, pvals, signifs)
kable(tbl, col.names = c("Perc. change", "P-value", "Significant?"),
      caption = "Coefficients as percentage change and their p-values")
```

Table 2: Coefficients as percentage change and their p-values

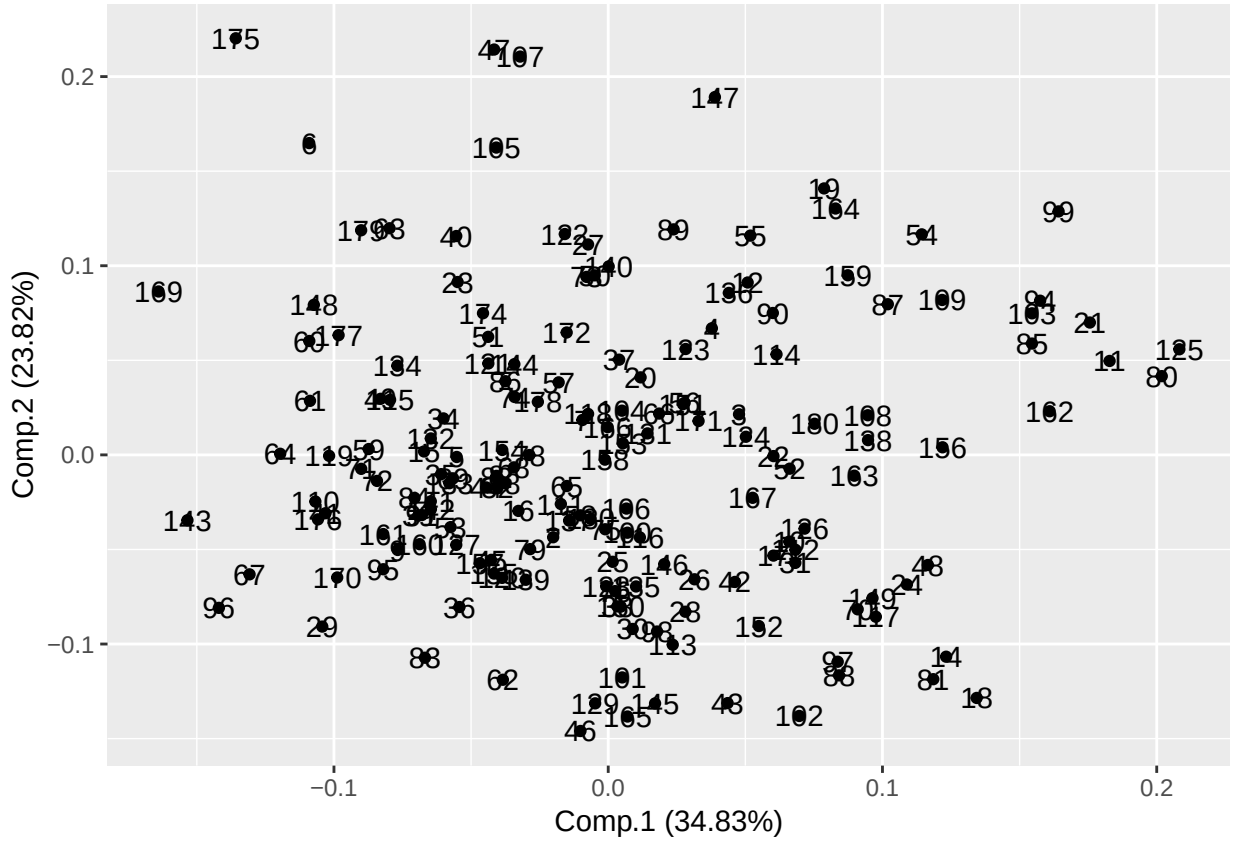
	Perc. change	P-value	Significant?
k_avg	-6.25	0.253	No
edge_length_avg	0.0158	0.845	No
intersection_density_km	0.754	0.031	Yes
street_density_km	-0.00196	0.35	No
self_loop_proportion	382	0.77	No
degree_centrality_avg	48400	0.928	No
clustering_coefficient_avg	-88	0.157	No
percentage_twoway	-0.582	0.002	Yes

From the above table it is seen that for each unit change in e.g intersection density, the average inner_time increases by 0.75%. Also for each percentage twoway roads added, the average "time it takes to travel one mile into the central business district during peak hours" is reduced by approximately 0.582%.

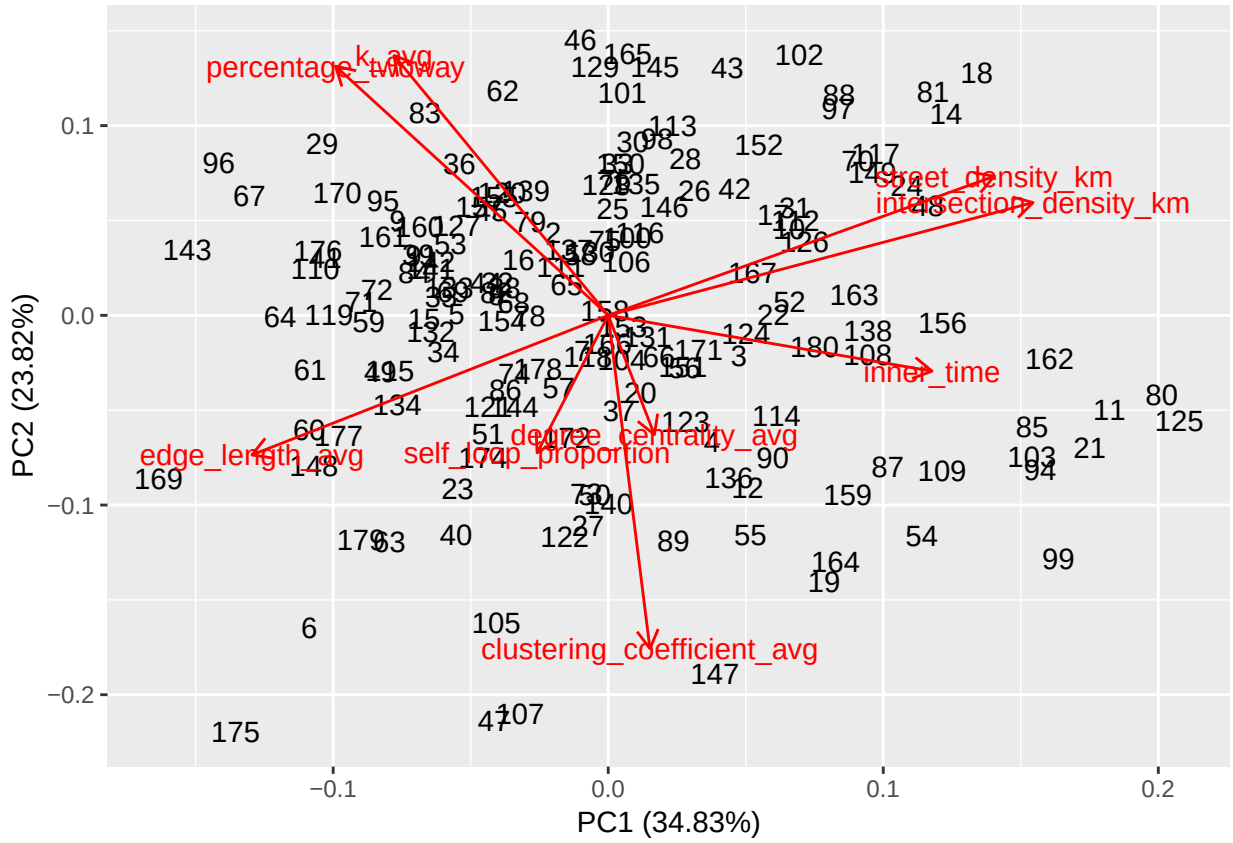
PCA

O

```
scaled_metrics <- scale(metrics[outliers_to_omit, regression_vars]) #  acapulco

metrics.pca <- princomp(scaled_metrics)

autoplot(metrics.pca, data = scaled_metrics, label=T)
```




```
met.pca <- prcomp(scaled_metrics)

autoplot(met.pca, data = scaled_metrics, label = T, shape = F, loadings = T, loadings.label = T)
```




```
summary(met.pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4     PC5     PC6     PC7
## Standard deviation     1.7706 1.4642 1.0570 0.9657 0.78357 0.71659 0.54687
## Proportion of Variance 0.3483 0.2382 0.1241 0.1036 0.06822 0.05706 0.03323
## Cumulative Proportion  0.3483 0.5865 0.7107 0.8143 0.88251 0.93956 0.97279
##                            PC8     PC9
## Standard deviation     0.45746 0.18870
## Proportion of Variance 0.02325 0.00396
## Cumulative Proportion  0.99604 1.00000
```