Assignment for the course '107.258 Computerstatistik'

WS 2022

The assignment needs to be returned as two files. A pdf in the form of a report explaining the analysis and the corresponding .Rmd file. Everything must be reproducible. Seeds are always your student number. The code should run **as is** on any computer.

The submitted documents should contain your name, student number and for which study program the credits should be registered. The resulting pdf should be a **self-contained report** with explanations in full sentences. Thus only relevant information should be printed and the pdf should not exceed 5-10 pages (e.g., no need to print data sets).

The assignment should be solved without help from others! If two reports are too similar both will be failed. Cheating is not tolerable.

The files should be submitted in TUWEL latest 11.12.2022.

Task 1 (25 points)

The data set `taxi_small.csv` in TUWEL contains information rides taken in a yellow taxicab in New York City in May 2015. The response variable of interest is the tip percentage, the amount tipped to the driver divided by the total fare amount. Relevant predictor variables include

- the number of passengers in the cab
- the hour of the pickup
- the day of the week of the pickup
- the neighborhood of the pickup (NTA codes - neighborhood tabulation areas - see https://www1.nyc.gov/assets/planning/download/pdf/data-maps/nyc-population/census2010/ntas.pdf)
- the neighborhood of the dropoff (NTA codes)
- possibly the total fare amount - maybe for smaller fares, the tip percentage is higher than for longer rides.

In order to obtain the sample you will be working on, do the following:

```
## df <- read.csv("taxi_small.csv")
set.seed(12345678) ## TO REPLACE WITH YOUR MATRIKELNUMMER
id <- sample(1:nrow(df), floor(0.8 * nrow(df)))
df <- df[id, ]
```

a. Preprocessing

- Check for missing values and duplicates and remove observations if necessary. Check the response variable for outliers. Remove the observations flagged as outliers in the boxplot.
- Make a new variable `tod` which transforms the pickup hour into a time-of-day factor with 5 levels: `morning` (between 06 and 10), `lunch` (between 11 and 15), `afternoon` (between 16 and 19), `evening` (between 20 and 22), `night` (between 23 and 05).

- Make a new variable `wday` which transforms the pickup weekday into a factor.
- Make new variables for pickup and drop-off locations which contain the borough instead of the neighborhood (only the first two letters of the NTA code).

b. Summary statistics

Generate tables in the report which contain the relevant univariate descriptive statistics for the pre-processed data.

If some factor levels contain less than 50 of the observations in your sample, you can merge them to the next smallest group iteratively until each level contains at least 50 observations.

c. Graphics

Generate appropriate plots to summarize and visualize the data. Focus on univariate plots of the response variable and on visualizing the relation between the response variable and the predictor variables.

Make sure that you also look at possible interactions of the predictors relate to the response variable. More specifically, it can be that the combination drop-off vs. pickup borough results in different tip percentages, than when looking at the drop-off and pick boroughs separately. Inspect this hypothesis in a visualization.

d. Linear model

Build a linear model using the predictor variables: number of passengers in the cab, total fare amount, time-of-day factor, weekday factor, pickup and drop-off borough and the interaction of the pickup and drop-off borough factors.

Compute and report the OLS estimates for the whole data set. You can use the default treatment contrasts for the coding of the factor variables.

Comment on the results and on the impact of the predictors on the response variable. (Focus on the direction of the relationship and the significance rather than the magnitude of the coefficients)

e. Interpretation of coefficients

Interpret the intercept, the coefficient of the passenger count and coefficient of the total fare amount.

f. Prediction

Predict the tip percentage for a one-passenger ride with pickup in Manhattan and dropoff in Brooklyn on a Monday morning, where the estimated total fare amount is 30 USD.

g. Model diagnostics

Inspect the model diagnostics resulting from `plot()`. Comment on the results. Which assumption is strongly violated?

h. Sparse OLS

Given the many factor variables, the model matrix for the model in d. can be rather sparse. Write a function `ols_sparse` which takes a sparse model matrix `X_sparse` and the vector `y` and computes the OLS solution. To create a sparse model matrix you can use:

```
X_sparse <- Matrix::sparse.model.matrix(formula, data = df)
```

You can compute the solution by the naive approach. Replace all matrix operations with the functions in the **Matrix** package which are designed for sparse matrices (e.g., `Matrix::crossprod`).

In `ols_sparse`, the user can also specify a further argument `scale_continuous_x` which contains a character vector which specifies the names of the continuous variables which should be scaled before estimation. E.g. if `scale_continuous_x = c("fare_amount")`, the variable `fare_amount` will be scaled by the mean and standard deviation and the corresponding coefficients will be the standardized ones. If `scale_continuous_x = NULL`, no variable will be scaled.

When implementing the solution, make sure that you reformulate the OLS problem such that you do not destroy sparsity whenever `scale_continuous_x` is not NULL. *Hint: you can compute the coefficients for the original variables and then transform in the end to the standardized ones.*

Call the function for `y <- df$tip_perc`, `X_sparse` where the formula is the one used in d. and `scale_continuous_x = c("fare_amount", "passenger_count")`.

i. Randomized test

Among the factor combinations of pickup borough and drop-off borough with at least 20 observations, pick the group with the smallest number of observations in your sample. For this group, fit a regression `tip_perc ~ fare_amount + passenger_count + wday + tod`. Check the normality of the residuals.

If the normality assumption is violated and the number of observations is small, it can be that the hypothesis tests are misleading. One could resort to using a randomization or permutation test for computing the p-values associated with the hypothesis tests to be employed. In the linear model, one strategy is to

- keep X as is,
- permute the y variable;
- re-estimate the models having the same structure as in d. using OLS on this permuted data and compute the test statistic using the estimated standard error.
- for m randomized such samples, compute the p-value.

(Note that this procedure is known as the Kennedy method and has certain drawbacks).

Implement such a procedure for the coefficient corresponding to `fare_amount`. At the top of your code set `set.seed(1234)`. Compare with the test you get in the standard `summary` output. Use $m = 1000$.

j. Student t regression

Write a function `negloglik_t_errors` which implements the negative log likelihood for the linear regression model under the assumption that the errors are Student-t distributed with mean zero, variance σ^2 and **known** degrees of freedom ν. The function takes as arguments `par`, the vector of coefficients β and σ which

should be estimated, the model matrix X, the vector y and the degrees of freedom ν as arguments *Hint: the log likelihood is*

$$\log\, p(y|X, \boldsymbol{\beta}, \sigma^2, \nu) = \sum_{i=1}^{N} \log\, p(y_i|x_i, \boldsymbol{\beta}, \sigma^2, \nu)$$

$$\begin{aligned}\log\, p(y_i|x_i, \boldsymbol{\beta}, \sigma^2, \nu) &= \log\left(\frac{\Gamma((\nu+1)/2)}{\Gamma(\nu/2)\sqrt{\nu\pi\sigma}}\left(1 + \frac{(y - X\beta)^2}{\nu\sigma^2}\right)^{-(\nu+1)/2}\right) \\ &\propto -0.5\log(\sigma) - \frac{(\nu+1)}{2}\log\left(1 + \frac{(y - X\beta)^2}{\nu\sigma^2}\right)\end{aligned}$$

Using built-in functions for general purpose optimization in R minimize this function to obtain the maximum likelihood estimates for the regression model in d. which now assumes t errors. Use the whole data set `df` in the computation and use as arguments `nu <- 3`, `X`is the model matrix in d. and `y <- df$tip_perc`.

Compare the estimates with the OLS solution. Discuss briefly the results.