Detecting Fraudulent Credit Card Transactions

Authors: Joseph Haaga and Christopher Broll
Group #: 6
Date: 06/24/2018

Introduction and Problem Motivation

We are interested in using *Bayesian* inference to explore the relationship between transactions on credit cards and instances of credit card fraud. We hope to find that the amount of the transaction can predict these instances with statistical significance. In essence, we expect to see a linear relationship between the transaction amount and the number of instances of credit card fraud. To validate our hypothesis, we first use *random forest classification* to measure the importance of the features in the dataset to confirm `transaction amount` as a significant predictor and then use *Markov Chain Monte Carlo* (*MCMC*) to estimate the posterior distributions of three bins of transaction amounts, which contain low, moderate and high transaction amounts. In the next Section, we discuss the data further and how we prepared these bins.

The Data

The dataset, which can be found on *Kaggle*, titled *Abstract Dataset for Credit Card Fraud Detection*, is a `csv` file that contains 3,075 transactions. Since the source of the data is unknown, we assume the dataset to be from a credible source and continue our analysis. The dataset includes the following 11 features:

Name	**Description**
`Merchant_id`	The unique ID of the merchant
`Average.Amount.transaction.day`	The average amount of the transaction per day
`Transaction_amount`	The amount of the transaction
`Is.declined`	Whether the transaction was declined or not
`Total.Number.of.declines.day`	Number of transactions declined at this merchant on day of this transaction
`isForeignTransaction`	Whether the transaction remained domestic or not

`isHighRiskCountry`	Whether the transaction took part in a country flagged as “risky”
`Daily_chargeback_avg_amt`	The daily average amount charged back to the credit card
`X6_month_avg_chbk_amt`	The 6-month average amount charged back to the credit card
`X6.month_chbk_freq`	The monthly average amount charged back to the credit card
`isFraudulent`	The target - whether or not the transaction was fraudulent

Method

Mentioned in the introduction of the report, we first use *Random Forest classification* to obtain the values of importance, a vector containing a measure of each feature’s ability to predict the target feature (in this case, fraud). Since it is a feature carrying significant importance, we then select `transaction amount` as our parameter.

Focusing on this parameter, we first dropped all observations where the `transaction amount` was 0, leaving us with 2984 of the original 3075 records. We then sorted the observations by `transaction amount` in ascending order. From there, we were able to split the observations into three lists of 994 transactions, where each list contains observations where the `transaction amount` falls within a particular range. An example is depicted below.

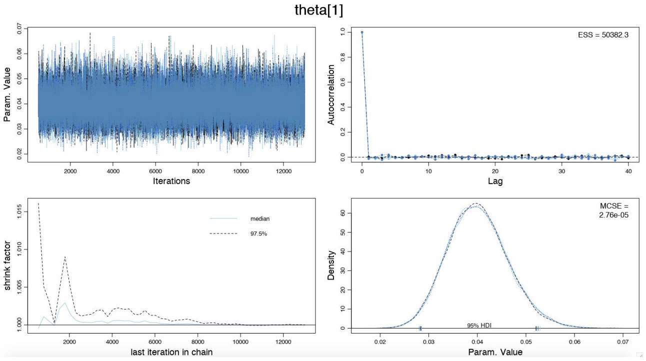
$0 < amount <= $3,974.215	$3,974.215 < amount <= $11,700.54	$11,700.54 < amount <= $80,000
Y	N	Y
N	N	Y
Y	Y	N
N	Y	N

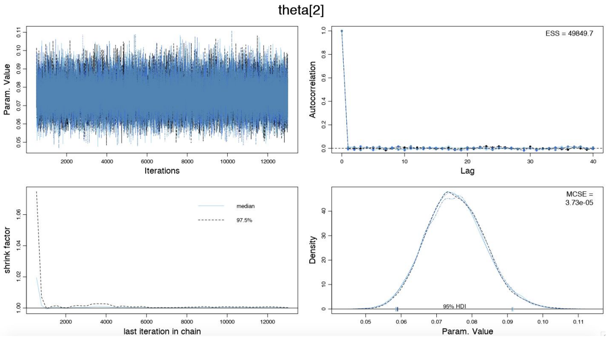
We then convert the Y/N factors into numerical values, and subtract 1 to binarize the values. With our data in the appropriate format, we can then pass the data frame, along with a model configuration, into *JAGS*. Using the *Exercise 2* boilerplate, we were able to generate an *MCMC*

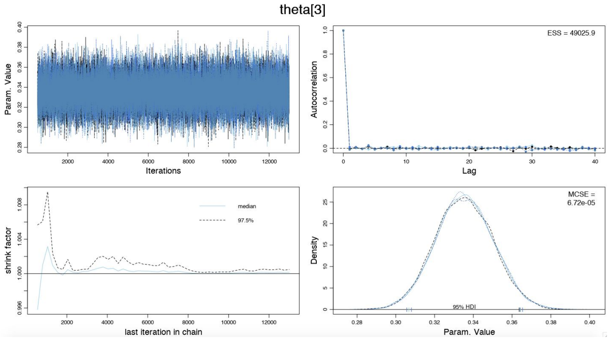
The *MCMC* approach seemed to fit the problem best, allowing us to observe and compare the posterior distribution of three different models. We were able to segment the dataset into three equal sized subsets by selecting records where the `transaction amount` is with a certain range. After running *MCMC*, we can observe the *HDI* for the posterior distributions, and determine whether there is a significant difference in the probabilities of fraudulent transaction for different transaction amounts.

Results, Analysis, and Further Research

Recall the three different bins that we created in the data preparation process, which we will call *bin-low*, *bin-medium* and *bin-high* (theta[1], theta[2] and theta[3], respectively). Aftering running the *MCMC*, we first take a look at the diagnostics for each respective bin.

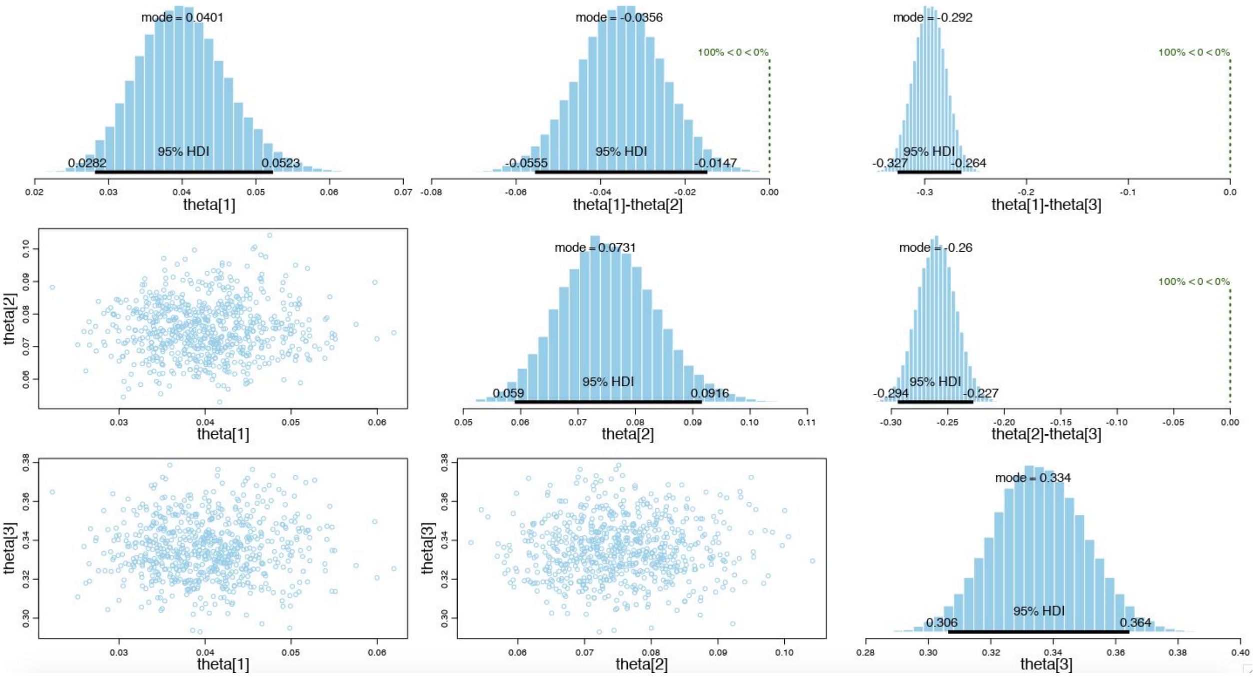






As you can see from the top-left panel, for the *bin-low*, the values for each chain remain in the same range of values after the *burn-in* period. The lower-right panel produces a density plot that shows much overlap between each chain, telling us that each chain is representative of the posterior distribution. The lower-left panel is a more direct diagnostic of convergence of the chains, measuring the average variance between all the chains and the chain itself. As a rule-of-thumb, as long as the last iteration reaches a shrink factor of 1, convergence is not an issue; hence, from the plot, we have convergence around iteration 8000. Finally, looking at the top-right panel, autocorrelation, which tell us the correlation between chain values and the chain values k steps ahead, i.e., the chain values compared to the chain values with a time lag. This gives us a measure of accuracy in the chain, requiring the correlation between the chain and the lagged chain to be close to 0 to be considered accurate, which is what is shown. Looking at the plots for *bin-medium* (theta[2]) and *bin-high* (theta[3]), we can see/draw similar results/conclusions.

Now that we have checked the diagnostics, we can then compare the differences between the parameters and estimate their posterior distributions (see figure below). The modes for theta[1], theta[2] and theta[3] are 0.0401, 0.0731, and 0.334, respectively. Comparing the parameters, in particular, theta[1] - theta[3] and theta[2] - theta[3], we get -0.292 and -0.26, respectively. In addition, since neither HDI contains 0, we can say with confidence that theta[1] and theta[2] are different than theta[3]; more specifically, since theta[3] contains the largest transaction amounts, this means that larger transactions are more likely to be fraudulent.



Although our hypothesis is confirmed, the mode 0.334 for theta[3] is not what we expected; instead, we expected to see a mode closer to 1, which indicates a fraudulent transaction. According to the *random forest classification*, transaction amount had the third highest importance, which can explain the mode on the left-side of 0.5 or perhaps we need to refine the bins more and isolate more extreme transactions on the high-end. It is also just as likely that there is not enough samples to represent the true distribution of fraudulent transactions, given transaction amounts. This leads us to suspect that perhaps our model needs to be extended to one of higher-order, meaning a *hierarchical model*. If we can find a dataset that includes not only a binary field of high-risk countries, which contained the most importance, but transaction amounts for high-risk countries, then maybe we could find a distribution that fits our expectations.