MEF UNIVERSITY

END of DAY CASH AMOUNT PREDICTION of ALL BANK BRANCHES

Capstone Project

Sefa Erbaş

İSTANBUL, 2018

MEF UNIVERSITY

END of DAY CASH AMOUNT PREDICTION of ALL BANK BRANCHES

Capstone Project

Sefa Erbaş

Advisor: Asst. Prof. Dr. Ahmet Serdar Tan

İSTANBUL, 2018

MEF UNIVERSITY

Name of the project: End of Day Cash Amount Prediction of All Bank Branches
Name/Last Name of the Student: Sefa Erbaş
Date of Thesis Defense: 10/09/2018

I hereby state that the graduation project prepared by Your Name (Title Format) has been completed under my supervision. I accept this work as a "Graduation Project".

10/09/2018
Asst. Prof. Dr. Ahmet Serdar Tan

I hereby state that I have examined this graduation project by Your Name (Title Format) which is accepted by his supervisor. This work is acceptable as a graduation project and the student is eligible to take the graduation project examination.

10/09/2018

Director
of
Big Data Analytics Program

We hereby state that we have held the graduation examination of _____ and agree that the student has satisfied all requirements.

THE EXAMINATION COMMITTEE

Committee Member	Signature
1. Asst. Prof. Dr. Ahmet Serdar Tan	……..
2. Prof. Dr. Özgür Özlük	……..

Academic Honesty Pledge

I promise not to collaborate with anyone, not to seek or accept any outside help, and not to give any help to others.

I understand that all resources in print or on the web must be explicitly cited.

In keeping with MEF University's ideals, I pledge that this work is my own and that I have neither given nor received inappropriate assistance in preparing it.

Sefa Erbaş 10/09/2018

EXECUTIVE SUMMARY

END OF DAY CASH AMOUNT PREDICTION
OF ALL BANK BRANCHES

Sefa Erbaş

Advisor: Asst. Prof. Dr. Ahmet Serdar Tan

SEPTEMBER 2018, 20 pages

Today, although there are different payment methods available, most of the daily payments are made in cash[1]. Accordingly, people's cash demand continues consistently. Due to its presence in most common networks, being as both supplier and custodian position of the cash money, the branches of the Bank are the most satisfying places that meet this need. In banks, the remaining cash at branches at the end of the day is idle money and leads to a potential loss of interest. The primary purpose of the banks is to leave cash at the branches at the optimum level and to obtain interest by depositing the excess cash to the Central Bank branches. The cash transfer is performed by outsourcing companies called CIT (Cash-in-Transit).

In this study, research was conducted to estimate branch end-of-day balances using different regression models. In order to reduce the effect of variance caused by the absence of any amount of limitation in transactions in branches, the daily estimation method has been adopted instead of the long-term estimation. The model has been developed for the estimation of the afternoon cash transaction amount of each branch based on before noon cash transaction details. The model output is grouped within a range of specific amounts, providing an output that determines the interval to take action for cash transfers.

[1] World Bank Group (2016). Cash vs. Electronic Payments in Small Retailing Estimating the Global Size *(Overview of Turkey)*

Key Words: Daily cash flow forecasting, Cash demand for branches, Cash optimization, Bank Branches

ÖZET

END OF DAY CASH AMOUNT PREDICTION
OF ALL BANK BRANCHES

Sefa Erbaş

Tez Danışmanı: Yard. Doç. Dr. Ahmet Serdar Tan

EYLÜL, 2018, 20 sayfa

Günümüzde, her ne kadar farklı ödeme yöntemleri mevcut olsa da, günlük ödemelerin yarısından çoğu nakit ile yapılmaktadır. Buna bağlı olarak insanların nakit gereksinimi sürekli olarak devam etmektedir. Yaygın ağlarda bulunması, hem tedarikçi hem de saklayıcı konumunda olması sebebiyle, bu ihtiyacı en çok karşılayan yerler Banka şubeleridir. Bankalar açısından bakıldığında ise, gün sonunda şubelerde kalan nakit, atıl para konumundadır ve potansiyel faiz kaybına yol açmaktadır. Bankaların öncelikli amacı, şubelerde optimum düzeyde nakit bırakmak, fazla olan nakdi Merkez Bankası Şubeleri 'ne yatırarak faiz elde etmektir. Bu dengenin sağlanabilmesi için, para nakil hizmeti veren firmalar ile çalışılmaktadır.

Bu çalışmada farklı regresyon modelleri kullanılarak, Şube gün sonu bakiyelerini tahmin etmeye yönelik araştırma yapılmıştır. Şubelerden gerçekleşen işlemlerde her hangi bir tutar sınırlaması olmamasının yaratacağı varyansın etkisini azaltabilmek amacıyla, uzun dönemli tahmin yapmak yerine, günlük tahminleme metodu benimsenmiştir. Her Şube özelinde, öğleye kadar yapılan işlemler baz alınarak öğleden sonra oluşabilecek nakit hareketlerinin tahmin edilmesi yönünde model geliştirilmiştir. Model çıktısı belirli tutarlar aralığında gruplanarak, hangi aralıkta nakit para nakli için aksiyon alınması yönünde çıktı sağlanmıştır.

Anahtar Kelimeler: Günlük nakit hareketleri tahmin etme, Şubelerin nakit ihtiyaçları, Nakit seviyesini ideal düzeyde tutma, Banka Şubeleri

TABLE OF CONTENTS

.

1. INTRODUCTION

In this assignment, the task is to predict end of the day cash balances in the branches so that redundant cash holding can be avoided, which would reduce loss of interest, and minimize probability of facing insufficient cash which may induce Bank with lower customer satisfaction. This increased efficiency could also result in higher profits for the Bank. For this purpose, relevant data *(daily deposit and withdrawn amounts, number of transactions, size of assets of Branches to differentiate them between each other)* at a branch level for 5 year-period had been gathered. The main concept of the modeling is to predict total deposit and withdrawn amount end of the day based on relevant data collected for before noon on the same day. In this way, money transfer requests can be made to the outsourced cash replenishment company on the same day for the branches which are detected by the model that their cash amount would exceed the pre-defined threshold.

1.1. A brief literature reviews

It is noted that most of the cash optimization and/or prediction studies have been performed for ATM. Dr. Justin Paul (2016) focus on the efficient functioning of the supply chain on information flow and the fund flow of the retail banks. According to his study, the main factors in the value chain are Cash flow, Information flow and IT infrastructure, delivery period of cash feed, Deposit and Receipts, Geographical locations and Status of accounts. He compared two Banks according to their inherent features and a time series analysis has been implemented. Simutis et al. (2008) present two different methods of forecasting the daily cash demand for automatic teller machines (ATM) which are flexible artificial neural networks (ANN) and support vector regression algorithms. In accordance with their purpose, they found that the flexible ANN produced slightly better results than the support vector regression algorithms. Wen-Hua Cui, (2014) analyzed four different methods *(the first moving average prediction method, the second moving average prediction method, the first exponential smoothing prediction and the second exponential smoothing prediction methods)* for improving the prediction accuracy of bank cash flow while implementing time series prediction.

Wagner (2010) provided an overview of the state of the art in research on competing techniques of forecasting daily demand in cash supply chains in order to determine the overall performance and the potential of joint forecasting for integrated planning. A vector

time series model and a seasonal ARIMA model as benchmarks have been implemented which has resulted in ARIMA has higher forecasting accuracy compared to the vector time series model. Canser BILIR (2018) analyzed an integrated cash requirement forecasting and cash inventory optimization model in both the branch and automated teller machine (ATM) networks of a mid-sized bank in Turkey to optimize the bank's cash supply chain.

1.2. About the Data

Daily before noon cash withdrawal and deposit data at a branch level over the period Jan 2013 to Mar 2018 has been gathered. There are 246 branches widespread all over Turkey and the dataset comprises number of 493.722 observation *(2007 day for each branch x 246 branches).* On the other hand, the observations of closed branches are also left in the data set in order to increase the model's learning ability by providing a greater number of observations.

There are 4 main items in the dataset:

- **Seasonal Factors:** Date, Day of the week, Day of the month, Week of the year, Month of the year, holidays, weekends.
- **Cash Transactions:** Number and amount of daily deposit and withdrawal for each branch occurred before noon.
- **Branch Detail:** Size of assets for each branch. This item gives information about each branch performance during the monthly period. On the other hand, it is also useful for separating branches between each other.
- **Macroeconomic Variables:** Daily exchange rate of dollar and the deposit interest rate information.
- **Target Variables:** Afternoon deposit and withdrawal amount for each branch and for each snapshot.

Data has been split into two parts as train and test. (%80 train, %20 test). On the other hand, the stratified sampling method has been implemented in order to add an equal amount of observation from each branch to the training data. After the model set up, the predicted afternoon deposit and withdrawal amount for each branch and each day will be produced as an output. Then this value will be compared with actual balance.

2. PROJECT STATEMENT AND METHODOLOGY

In this section, the project statement will be detailed with defining the main problem. On the other hand, project scope and methodology will be enlightened.

2.1. Problem Statement

In recent years, although there has been a significant growth in the prevalence of credit cards, electronic transactions and other new payment instruments, the use of cash is still more important in the daily economy. Thus, bank branches play an important role in meeting the demands of this cash cycle management at all times.

Hundreds of transactions are made during the day in the Bank's branches. Generally, many of these transactions would be cash deposits and withdrawals. Because of these cash flow transactions, a cash balance occurs at the end of each day in each branch. Banks do not want to hold a high level of cash balance at the end of the day. Because the remaining cash in the branch is a kind of idle and leads to a loss of potential interest. On the other hand, the banks can earn daily interest as much as the amount they deposit on the Central Bank. These money transfers are made by outsourcing companies which provides cash transportation services. However, there is a fee for this kind of transferring services, and it is also necessary that the transfer requests have to be entered into the system priory for a certain period of time during the day. It means, the transfer service cannot be requested at the end of the day.

Currently, a daily upper limit for each branch was set by Head Office and cash handling management was left to the Branch initiative. Branches are exposed to the penalty if they exceed their limits at the end of the day.

If we formulate the end-of-the-day cash balance;

End of the Day Cash Balance	=	**Cash Balance of Previous Day**	+	**Total Deposit Amount of the Day**	-	**Total Withdraw Amount of the Day**

With the model, it is possible to keep the end-of-day cash balance at the optimum level by estimating the total withdrawn and deposited amount in the day on a branch basis.

2.1.1. Project Objectives

The main objective in this project can be defined as minimizing the excess cash in bank branches by keeping the cash in the right place at the right time without a decrease in customer satisfaction levels because of shortage of cash.

Basically, the ways to be followed in order to provide optimized day-end cash balance in the branches can be summarized as follows:

1. Predicting daily cash withdrawal and cash deposit amounts in the branches
2. Minimize the end-of-day cash balance subject to the minimum service requirements.

2.1.2. Project Scope

This project focus on predicting daily deposit and withdrawal amount made in branches. While gathering data set, the ATM's which are located in branches have also taken into consideration, because their transactions also effect directly the end of day cash balance. Although the branches are closed on holidays, because of ATM transactions, there is also same variables in the dataset for holidays. However, this information is only used as an input for the model, ATM's cash balance prediction is out of scope for this assignment.

On the other hand, because of the large cash fluctuation occurs in branches, cash optimization research will not be performed for minimizing cost, maximizing profit, and improve customer satisfaction constraint. Nevertheless, break-even point approach could be proposed for optimization. The potential interest return of model output end of day cash balance amount for all branches is calculated. If this amount would be higher than the cash transfer cost, a request is made to CIT Company to collect the cash from the branches.

2.2. Methodology

During the data collection phase, two assumptions have been adopted to set the model. One of them is setting two model instead of one. The other one is predicting afternoon cash balance based on before noon transaction size. These assumptions have been detailed below:

2.2.1 Predicting the deposit and withdrawal amounts with separate models.

At the end of the day, the cash balance amount is a most important thing for the purpose of this study. Because, if this amount would be greater than the predefined threshold,

it is requested from the CIT (cash-in-transit) company to collect the excess cash amount from the relevant branch. However, end of day cash balance consists of two main items. One of them is deposit and the other one is withdrawal and the cash balance is calculated as *(deposit – withdrawal).* Instead of setting one model which the target variable is end of day cash balances in the development dataset, it is decided to set two model with same dataset, one of them predicts deposit amount and the other one predicts withdrawal amount. This approach would provide the model be able to capture the differentiation of the branches according to the transaction size. *(Let's assume that we have two branches and their end of day cash balances are almost identical on the same snapshot date. However, the transaction size could be more different than each other. Possible transaction amounts for both branches have been stated below.)*

Table 1. Possible End of Day Cash Balance Amount for 2 Different Branch

Same Snapshot	Branch "A"	Branch "B"
Deposit Amount	1,000,000.00 ৳	150,000.00 ৳
Withdrawal Amount	950,000.00 ৳	100,000.00 ৳
End of day Cash Balance	**50,000.00 ৳**	**50,000.00 ৳**

2.2.2 Afternoon cash balance

Due to daily large cash fluctuations occur in the branch side, it is hard to make a long-term cash balance prediction such as weekly or monthly. Because customers have no daily amount limit to make deposit or withdrawal like being on ATMs. That's why it is decided to make daily prediction. On the other hand, it is noted that daily number of transaction information is also important to determine the end of day cash balance. In the light of these, the dataset has been built based on before noon transaction size and the targets are afternoon deposit and withdrawal amount.

After the branches give lunch break, the before noon data is extracted from the system and is given to the model as an input. According to model results, the branches that would exceed the end-day cash balance limit or would suffer insufficient cash balance will be determined. Hence, the whole cash transfer planning can be made during lunch break.

2.2.3 Model Selection

Data gathering process had been started from SQL Developer with accessing directly to Bank server. After the raw data is gathered, the exploratory data analysis is performed with IDEA CaseWare software. Then the final data has been uploaded to Microsoft Azure ML Studio and all model setting and fine-tuning process had been performed in this platform. Four different models which are stated below have been executed.

Random Forest Regression

Decision tree algorithm breaks down a dataset into smaller subsets with trying to homogenize each decision node and it is finalized when terminal node is generated for each leave. At the end of the process associated decision tree is incrementally developed. Decision tree regression is trained on a series of examples with outcomes in a continuous range. These training examples are partitioned in the decision tree and the new instances that end on a specific node and will get the average of the training example values. Besides, random forest regression is a regression model that makes accurate estimates by generating more compatible models using more than one decision tree.

Boosted Decision Tree Regression

Boosting refers to creating multiple decision trees which are dependent on prior trees. The first decision tree would try to predict the target variables while the other trees attempt the difference between the actual values of the target variables and predicted by the previous tree. That's why it is also called as ensemble methods which correspond to combining several decision trees to produce better predictive performance compared to a single decision tree.

Bayesian Linear Regression

In Bayesian model, simple linear regression is calculated using probability distributions, not the exact points. The target value is not predicted as a single point, there is an assumption for the target values that it would be drawn from a probability distribution. The substructure of the distribution is formed from normal (Gaussian) Distribution compromised by a mean and variance.

Linear Regression

Linear regression is a very simple approach in supervised learning methods that basically is used to predict a quantitative output from the values of input variables. The assumption of the model is there is a linear relationship between input(s) and output.

2.3. Data Analysis

Accounting refers to a set of rules that record, classify, summarize and report the financial transactions of all organizations in a monetary statement and interpret the results. These rules also apply to the Banks and are standard for every Bank in the world and it is called as "Bank Accounting." Every transaction must be recorded and kept in Bank system in line with these rules.

Hundreds of transactions like issuing a credit, money transferring, clearing the debt, cash deposit etc. are performed in Banks every day. Every type of transactions has special accounting code. The codes of some Banking items have been stated as below for instance.

Table 2. Example of Some Banking Accounting Codes

Accounting Code	What does the code correspond to?
010 00 0 0	Cash *(Turkish Lira)*
118 01 0 0	Short-Term, Secured Revolving Credits
220 01 3 3	Accrued Interest on Credit Cards
311 00 0 0	Fixed Rate Deposit Account
420 02 1 0	Distributable Profits
535 24 0 0	Interest from Export Loans
760 05 0 0	Banking Service Revenues- Transfer Commission
850 01 0 0	Real Estate Depreciation Expense

In the Bank Accounting system, transactions related to each other *(labeled as debit and credit)* are recorded mutually. For example; if a client withdrawn money *(1,000.00 ₺)* from his/her deposit account, the transaction is registered as;

Table 3. Accounting Records

Account Name	Debit	Credit
Deposit Account	1,000.00 ₺	
Cash		1,000.00 ₺

Conversely, if a client deposit money to his/her account;

Account Name	Debit	Credit
Cash	1,000.00 ₺	
Deposit Account		1,000.00 ₺

In this project, only the activities in the "cash" item will be examined. Because, the main aim of the project is predicting end of day cash balance after the all this kind of transactions have been made during all day in the branches.

There is a table in data warehouse of Bank which keeps the all transactions with more detail information. This information has been stated below:

- Date of Transaction
- Time of Transaction
- The user code of the staff who performed the transaction
- Amount
- Debit / Credit information
- Currency Type
- Receipt Number
- Branch Code
- Accounting Code
- Type of Transaction *(Bank main system modules that is used in the transaction)*
- Customer Number
- User Explanation of the Transactions

For the purpose of this project, the data have been extracted with an aggregation format *(sum of amounts)* based on each day, branch and debit/credit type. As it is seen from the script, some transaction types have been excluded while the data was gathering. These transaction types are:

- **Cash Transportation Transactions:** In current circumstance, the operation staff manages the cash handling process in Branch manually. If he/she foresees that there would be need of more cash or would be excess cash during the day, he/she gives a request to CIT (Cash-in-transit) company to collect more cash or hand over the excess cash. This kind of transactions are also recorded and kept in the table. They are excluded because, it has been solely focused on the amount of money deposit from / withdrawal made by customers. These are the two main items that affects the remaining balance in the branches. After all, the purpose of this project is to determine how much of the money transfer will be requested to/from the CIT company after predicting the end-of-day cash balance.

- **Cash transfer transactions which are made between tellers in branch:** There are 3 type of vault in the branches. One of them is main vault, one of them is ATM's vault and the other one is staff's *(tellers & operation responsible)* vault. There are so many transfer transactions between these vaults during the day. On the other hand, there is also a rule that, all cash must be recorded under the main vault at the end of the day. These transactions do not generate real cash fluctuations, that's why they are excluded.

- **Outlier transaction amounts:** As there is no limit for the transactions made in branches, it is noted that there are some transactions that were made exceedingly above the overall average. In fact, these types of transactions are carried out by notifying the relevant branch in advance. Most of customer is aware of the cash balance problem in the Branches, and if they would like to withdraw a high amount of money, they inform the relevant Branch at least one day earlier. However, it cannot be stated that this practice also applies to the money deposit process. Nevertheless, that is a well-known problem that, the outliers may change the model equation completely. That's why they are also excluded from the main data.

The main data of this project comprises two hierarchic table. The first one is the raw table that the all transactions are recorded one by one. The other one which is the main input for the model has been aggregated for each branch and each day with totalizing the withdrawal and deposit amounts. Outlier analysis had been performed on both tables to be able to keep more observations for the model input. The percentile ranks are stated below.

Table 4. Percentile Ranking of Raw and Main Tables

Percentile	Raw Table		Aggregated Table (Model input)	
	Before noon Amount	Afternoon Amount	Afternoon Deposit	Afternoon Withdrawal
Min	0.01 ₺	0.01 ₺	0.00 ₺	0.00 ₺
0.5th	1.00 ₺	1.00 ₺	1,100.00 ₺	2,550.00 ₺
1st	5.00 ₺	3.00 ₺	8,065.00 ₺	8,500.00 ₺
25th	100.00 ₺	102.00 ₺	129,238.00 ₺	111,012.00 ₺
75th	995.00 ₺	880.00 ₺	328,036.00 ₺	283,957.00 ₺
95th	6,000.00 ₺	5,300.00 ₺	564,085.00 ₺	484,250.00 ₺
99th	31,000.00 ₺	39,000.00 ₺	784,196.00 ₺	676,276.00 ₺
99.5th	53,000.00 ₺	55,000.00 ₺	875,567.00 ₺	757,394.00 ₺
99.9th	150,000.00 ₺	152,000.00 ₺	1,085,024.00 ₺	947,231.00 ₺
Max	35,102,311.00 ₺	440,000,000.00 ₺	2,043,510.00 ₺	3,018,781.00 ₺

For the raw table, only greater than 99.9^{th} percent values had been excluded while the aggregated data was executed within SQL Developer. On the other hand, for the model input data, lower than 0.5^{th} percent and the greater than 99.9^{th} percent variables had been excluded to avoid adverse effects of the regression model equation.

After that execution of the query, the total amount of money withdrawals and deposited and their counts *(from 01:00 am to 1:00 pm – according to methodological assumption stated in the title of "Methodology")* from 245 branches was obtained on a daily basis with a length of 5.5 years.

In the first raw form of data, the total withdrawal and deposited amount and their counts for a branch on a given day comes in two different rows. A unique key was generated for each branch and also each day by the combination of branch code and date. Then the deposit and withdrawal amount information of each branch and each day have been gathered to the same rows with using the unique key.

The report of Branch's asset size has also been extracted from the Banking system. This report includes monthly asset size information of each branches. This information has also been merged with the main data for each branch of each month.

Then seasonality variables had been created based on the date:

- Which day of the week information *(Monday: 1, Tuesday: 2, Wednesday: 3, Thursday: 4, Friday: 5, Saturday: 6, Sunday: 7)*
- Which day of the month information *(1, 2, 3, 4….,29, 30, 31)*
- Which month of the year information *(1, 2, 3. …., 10, 11, 12)*
- Which week of the year information *(1, 2, 3, 4….,50, 51, 52)*

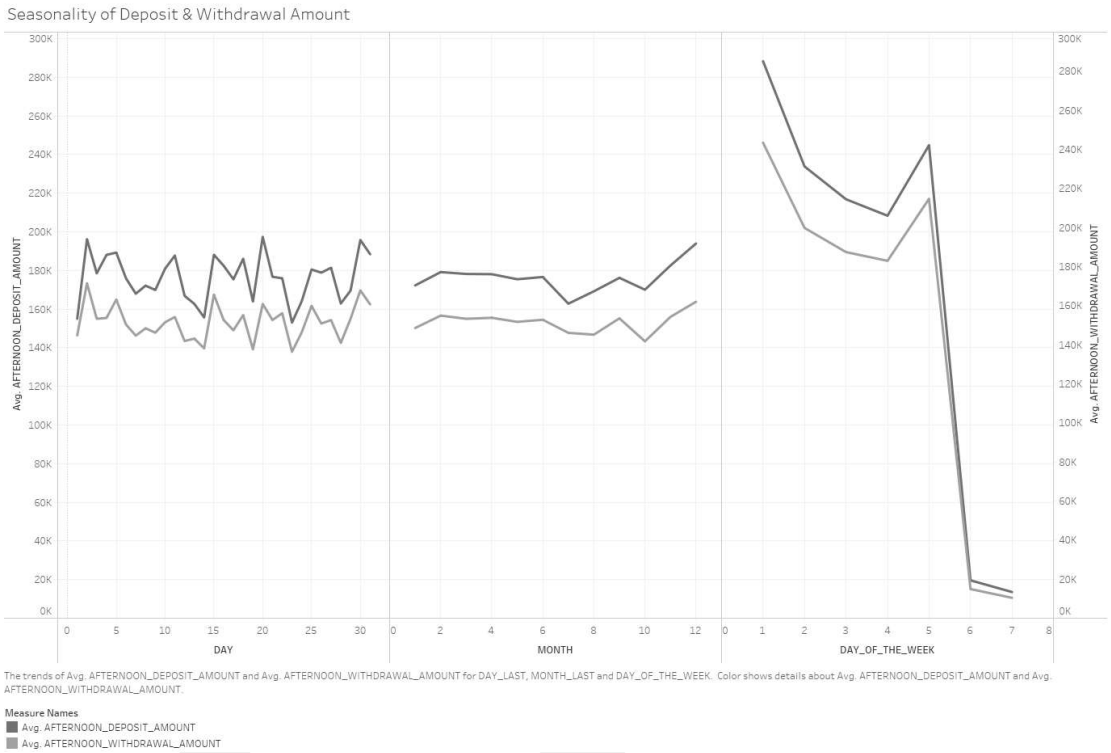
Then a binary column had been created which includes the holiday information. If the relevant date is a holiday the value is "1", if not which means it is a working day, the value is "0".

Lastly, the macroeconomic variables *(FX Rate (Daily Range) & Interest Rate (Monthly Range))* had been combined with the main data.

Finally, same query had also been executed again, just changing the time information *(from 1:00 pm to 11:00 pm – according to methodological assumption stated in the title of "Methodology")* to determine the target variables which are afternoon deposit and withdrawal amount. After getting the raw data, similar modifications stated above have also been implemented. Then this amount information has been merged with the main data.

At the end of the data gathering process, it is seen that, there are some missing values for some day and some branches based on deposit and withdrawal amount. When a query was extracted to find out the reason which includes all transactions detail on the related day, it is seen that, there is no relevant transactions at all on the specified date and/or time range. That's why, the missing values has been replaced by "0".

Graph 1. Daily & Weekly & Monthly Distribution of the Deposit and Withdrawal Amount of Input Data



As it is seen in the graph that, the difference in average transaction amounts between workdays and weekends are very huge and it is also valid for holidays. For that reason, holidays are not included in the final input data *(the model training data)* to reduce the variance of a week cycle.

3. RESULTS

All model setting had been performed in Microsoft Azure Machine Learning Studio. After gathering the last form of the data, the basic steps such as selecting relevant observations and features, data manipulation operations, creating dummy variables for the categorical features, splitting the data for test & train, selecting the related regression model, training the model, implementing the model equation on the test data and finally collecting the results had been performed.

As it is mentioned before, two different model had been established with using same dataset; one of them is for predicting the afternoon deposit amount of the branch for each day, and the other one is predicting the afternoon withdrawal amounts in same conditions. Four different regression model *(Decision Forest Regression, Boosted Decision Tree Regression, Linear Regression and Bayesian Linear Regression)* was used for the prediction. As it is seen below, the performances of all the models were close to each other according to "Root Mean Square Error[1]" values. For that reason, ensemble averaging method had been implemented by taking the average of the prediction of the all model outputs. The aggregate opinion of multiple models would be less noisy than other individual models. The model results are stated below:

Table 5. Detail of Model Performance Metrics

Model	Algorithm	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Afternoon Deposit	Decision Forest Regression	81,471.02	108,001.63	0.66	0.45	0.54
	Boosted Decision Tree Regression	75,349.24	100,184.64	0.60	0.38	0.61
	Linear Regression	79,551.94	105,608.67	0.64	0.43	0.56
	Bayesian Linear Regression	79,560.32	105,624.33	0.64	0.43	0.56

[1] Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit. (http://www.statisticshowto.com/rmse/)

Model	Algorithm	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Afternoon Withdrawal	**Decision Forest Regression**	77,226.15	102,324.32	0.72	0.54	0.45
	Boosted Decision Tree Regression	73,071.92	96,891.10	0.68	0.48	0.51
	Linear Regression	75,609.88	100,382.84	0.70	0.52	0.47
	Bayesian Linear Regression	75,629.38	100,413.81	0.70	0.52	0.47

According to this project approach, the ordinary performance outputs are not used directly for evaluation metrics. The important thing in this approach is not the individual performance criteria of each model *(deposit and withdrawal)*, however how the subtraction of the outputs *(which absolutely gives the end of day balance that desired to find out)* differ from the real amount is matter. It means the predicted balance should be compared with the real end of day balance and it gives the real performance level of the models.

For that reason, in order to be able to provide real performance measurement, the outputs produced by the models were subtracted from each other for each observation placed in test data to find the predicted balance. Then, the predicted balance was also subtracted from the real balance amount to find out the variance between them.

The absolute of the subtracted values have been splatted into bins with 100,000.00 ₺ intervals. It is learned from the Bank Cash Handling Department that, 100,000.00 ₺ is the maximum tolerable point to keep this additional amount in the Branch. It can be also stated that, this amount (100,000.00 ₺) is breakeven point for the cash transfer decision; the cost of the cash holding, and the cost of the transportation is exactly same at that point.

According to this performance metrics, the outputs of the model is stated below.

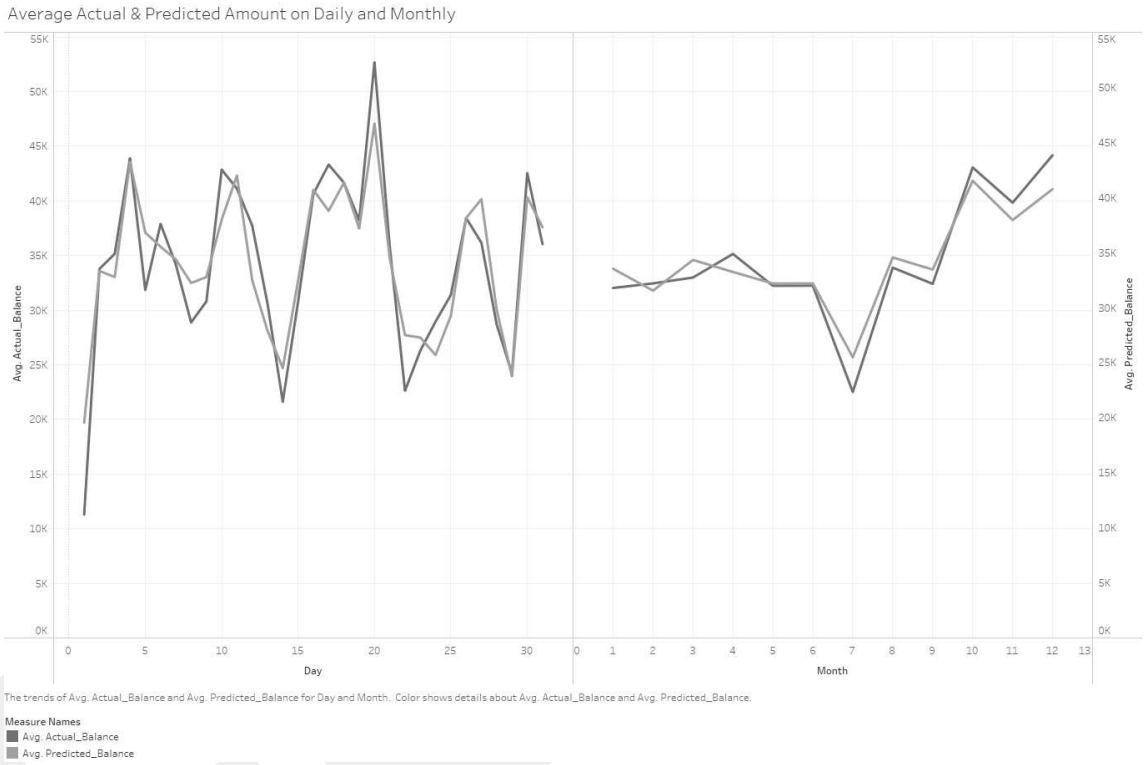
Table 6. Detail of Model Performance Metrics on Predicted Balance

Category	#Number	Percentage
0 - 100K	47,951	64.63%
100K - 200K	18,620	25.10%
200K - 300K	7,620	10.27%
Total (Test data observation)	**74,191**	**100.00%**

According to this model outputs, %65 ratio of end of day cash balance had been predicted in tolerable interval (0 – 100K). It also means that, with implementation of the model, 65% of idle money will be used efficiently.

On the other hand, when the predicted & actual balance amounts are compared on monthly and daily average, it has seen that, their values are very close to each other. Model is able to capture the seasonality effect on the cash fluctuations.

Graph 2. Average Actual & Predicted Amount on Daily and Monthly



4. DELIVERED VALUE AND FURTHER STEPS

4.1. Project's Delivered Value & Social and Ethical Aspects

In daily life, all of activities repeats itself with a certain cycle. We can only capture a small part of this loop as a human. Nowadays, if we have enough data includes all detail transactions/ activities of the routine cycle we can find out the secret patterns in it.

In this project, it is tried to predict the end of day cash balance of the Bank Branches to provide optimized and effective cash handling management. Effective cash management helps to meet customer needs on the one hand and contribute positively to the profitability of the banks on the other hand.

4.2. Future Research

In this project, the scope involves only predicting end of day cash balance of Bank branches and excluding ATM's transactions. Although some ATMs are located in branches and their transactions had been covered in the current data indirectly, there are so many ATM located out of the branch. Based on cash demand forecasting an intelligent cash management system can provide the bank reduce its operational expenses and improve the return on its cash assets. It is planned to conduct a research involving all parties *(Banks, CIT companies, Central Bank*) that need to effectively manage their cash demand.

5. REFERENCES

[1] Simutis, Rimvydas, Dilijonas, Darius & Bastina, Lidija (2008). Cash Demand Forecasting for ATM Using Neural Networks and Support Vector Regression Algorithms. 20th EURO Mini Conference "Continuous Optimization and Knowledge-Based Technologies" *ISBN 978-9955-28-283-9*

[2] Paul, Justin (2016). ATMs and Cash Demand Forecasting: A Study of Two Commercial Banks

[3] Cui, Wen-Hua, Wang, Jie-Sheng & Ning, Chen-Xu (2014). Time Series Prediction Method of Bank Cash Flow and Simulation Comparison. *ISSN 1999-4893*

[4] Wagner, Michael (2010). Forecasting Daily Demand in Cash Supply Chains. American Journal of Economics and Business Administration. *ISSN 1945-5488*

[5] Ashokkumar, Kirthika, D., Thinesh, & Dr. Srinivasakumar (2016). A Survey on Cash Demand Forecasting for ATM's Using Different Financial Modelling Techniques. *ISSN 2321 – 919X*

[6] Ray, Sandipan, (2010). Determining Optimal Cash Allocation at ICICI Bank Branches with SAS® Enterprise Guide® and SAS/OR® Software. *SAS Global Forum 2010 Paper 239*

[7] Darwish, Saad M., (2013). A Methodology to Improve Cash Demand Forecasting for ATM Network. International Journal of Computer and Electrical Engineering. *10.7763/IJCEE. 2013.V5.741*

[8] Bilir, Canser & Doseyen, Adil (2018). Optimization of ATM and Branch Cash Operations Using an Integrated Cash Requirement Forecasting and Cash Optimization Model. Business & Management Studies: An International Journal. *pp. 237-255*

[9] Batı, Seyma & Gözüpek, Didem (2017). Joint Optimization of Cash Management and Routing for New-Generation Automated Teller Machine Networks. *2168-2216*

[10] Cabello, Julia García (2013). Cash Efficiency for Bank Branches. García Cabello SpringerPlus, *2:334*

APPENDIX A

MODEL LIFE-CYCLE PROCESS

- It starts with data gathering process from Bank's data warehouse via SQL Developer. To find out the outliers from the transactions amount, this script stated below is used. It returns the transaction amount and its percentile ranking.

```
SELECT AMOUNT, PR FROM (SELECT AMOUNT,
PERCENT_RANK()
OVER (ORDER BY AMOUNT ASC) AS PR
FROM DAILY_TRANSACTIONS
WHERE TRANSACTION_DATE >= TO_DATE ('2013-01-01', 'YYYY-MM-DD')
AND TRANSACTION_DATE <= TO_DATE ('2018-06-30', 'YYYY-MM-DD')
AND TRANSACTION_TIME BETWEEN '00:01:00' AND '13:00:00'
AND TRANSACTION_TYPE NOT IN ( 'ABC123', 'CED107' , 'DEF108')
AND TRANSACTION_STATUS <> 'CANCELLED'
AND (ACCOUNTING_CODE= '100000' ))
WHERE PR < '0.999'
ORDER BY PR DESC
```

- After, defining the outlier amounts, new script had been executed to aggregate the daily transactions in terms of each day and each branch with summing up the transactions amount.

```
SELECT * FROM
(SELECT BRANCH_CODE, ACCOUNTING_CODE, TRANSACTION_DATE
SUM (CASE DEBIT_CREDIT
WHEN 'DEBIT' THEN – AMOUNT
ELSE AMOUNT
END) AS T_ AMOUNT, DEBIT_CREDIT, COUNT(*) AS ADET
FROM
(SELECT *
FROM DAILY_TRANSACTIONS
WHERE TRANSACTION_DATE >= TO_DATE ('2013-01-01', 'YYYY-MM-DD')
AND TRANSACTION_DATE <= TO_DATE ('2018-06-30', 'YYYY-MM-DD')
AND TRANSACTION_TIME BETWEEN '00:01:00' AND '13:00:00'
AND TRANSACTION_TYPE NOT IN (‘                ABC123', 'CED107' , 'DEF108')
AND TRANSACTION_STATUS <> 'CANCELLED'
AND AMOUNT <= ‘150000’
AND (ACCOUNTING_CODE= '100000’))
GROUP BY BRANCH_CODE, DEBIT_CREDIT, ACCOUNTING_CODE, TRANSACTION_DATE)
WHERE T_ AMOUNT <> 0
ORDER BY TRANSACTION_DATE ASC
```

- After getting the raw data, exploratory data analysis had been performed in IDEA Caseware. *(Missing value treatment, joining with the other datasets includes the other variables, manipulating the data)* The final model input data had been prepared.

- The final data had been uploaded to 'R' to detect whether there are any outlier values in the target variables *(afternoon deposit & withdrawal amount for each day and each branch)*

#Outlier Detection in Aggregated Table (Input Data)

```
#Afternoon Deposit Amount Variables
quantile(model_datalast$
AFTERNOON_DEPOSIT_AMOUNT, c(.005, .01,.25,.75, .95, .99, .995,.999))
'   0.5%       1%         25%         75%         95%         99%       99.5%       99.9%
 1100.00    8065.9    129238.0    328036.2    564085.9    784196.2    875567.4   1085024.4 '

#Min - Max
summary(model_datalast$AFTERNOON_DEPOSIT_AMOUNT)
'   Min.    1st Qu.     Median      Mean     3rd Qu.     Max.
      0     129238     213520     247526     328036    2043510 '


#Afternoon Withdrawal Amount Variables
quantile(model_datalast$
AFTERNOON_WITHDRAWAL_AMOUNT, c(.005, .01,.25,.75, .95, .99, .995,.999))
'    0.5%       1%         25%       75%         95%         99%       99.5%       99.9%
   2550.0    8500.0    111012.6   283957.1    484250.3    676276.8    757394.9    947231.5 '

#Min - Max
summary(model_datalast$AFTERNOON_WITHDRAWAL_AMOUNT)
'   Min.    1st Qu.    Median      Mean     3rd Qu.      Max.
      0     111013     184931     213578     283957     3018781 '
```

- Then, the final data had been uploaded to Microsoft Azure Machine Learning Studio and these steps stated below had been performed respectively:

    - **Apply SQL Transaction:** Only workday observations had been selected and the observations which has outlier values had been excluded.

```
select * from t1 where holiday='0'
and AFTERNOON_DEPOSIT_AMOUNT between '1100' and '1085024'
and AFTERNOON_WITHDRAWAL_AMOUNT between '2550' and '947231'
```

- **Select Columns in Dataset:** Necessary features has been selected to make input for the model.
- **Edit Metadata:** Required column type had been transformed to categorical variables such as "Branch Code, Date information *(Day & Month)*"
- **Convert to Indicator Values:** Dummy variables had been created for the categorical features.
- **Normalize Data:** There are some numeric features from different scale and they are also skewed distributed. "Lognormal" transformation method had been applied to make the distribution normally.
- **Split Data:** The transformed data had been splatted as a test and train *(%80 train, %20 test)*
- **Selecting the Algorithm:** 4 different model algorithms had been selected to set up.
- **Train Model:** The data has been trained with relevant algorithm with a default parameter. In fine tuning stage, too many combinations of parameters had been executed with "Tune Model Hyper parameters" module, however, the best "RMSE" value had been gained from default parameters.
- **Score Model:** After training, each algorithm had produced prediction values for the test data.
- **Evaluate Model:** The performance metric *(Mean Absolute Error, Root Mean Squared Error, Relative Absolute Error, Relative Squared Error, Coefficient of Determination)* values had been checked.
- **Cross-Validate Model:** The whole dataset had been split into 10 subset dataset *(fold)* and each model had been generated for each subset. According to accuracy statistic, it is noted that the standard deviation *(between the Coefficient of Determination scores of each fold amongst same model)* of each model type ranges from 0.0025 and 0.0052.
- **Join Data:** Each predicted value from the 4 different algorithms had been gathered together for each model *(Deposit & Withdrawal)* and their averages had been calculated for each observation placed in test data. After all this calculation, we got the final predicted deposit and withdrawal amounts.
  Then, the withdrawal amount was subtracted from the deposit amount for each observation to find the remaining balance. Then, the predicted balance was also subtracted from the actual balance to detect how much the model result differs from the actual balance.
- **Group Data into Bins:** The variance between the actual and predicted balance amount had been splatted into bins to be able to measure to model performance easily.

[1] The table and variable names in SQL scripts had been manipulated because of the privacy rules.

The process flow performed in Microsoft Azure ML Studio is stated below: