www.ceng.deu.edu.tr

DOKUZ EYLUL UNIVERSITY ENGINEERING FACULTY

DEPARTMENT OF COMPUTER ENGINEERING

CME-4423 MOBILE DEVICES AND APPLICATIONS

DECISION SUPPORT SYSTEM

PROJECT REPORT

by

Ezgi BOZDEMİR 2012510010

ezgi.bozdemir@ceng.deu.edu.tr

Azime Özge KURAL 2012510044

azime.kural@ceng.deu.edu.tr

8.12.2016

İZMİR

Table of Contents

1. INTRODUCTION

In this report we have done the example of web services. We have did the web service example to use in the project. Our aim is to write web service and attract data. We will use any database for this.

2. PROJECT DESCRIPTION

Due to the increase of data generation rates nowadays, analysis of this data is more complex than ever. To make a sense from this data there is a need of a system. With this study we are aiming to develop an intelligent decision support system based on clustering and association techniques.

3. PROGRESS SUMMARY

3.1. FOURTH WEEK

3.1.1. DECISION SUPPORT SYSTEMS

What is Decision Support System?

(DSS) are a specific class of computerized information system that supports business and organizational decision-making activities. A properly designed Decision Support System is an interactive software-based system intended to help decision makers compile useful information from raw data, documents, personal knowledge, and/or business models to identify and solve problems and make decisions.

Typical information that a decision support application might gather and present would be:

- Accessing all of your current information assets, including legacy and relational data sources, cubes, data warehouses, and data marts
- Comparative sales figures between one week and the next
- Projected revenue figures based on new product sales assumptions
- The consequences of different decision alternatives, given past experience in a context that is described

Information Builders' Web FOCUS reporting software is ideally suited for building decision support systems due to its wide reach of data, interactive facilities, ad hoc reporting capabilities, quick development times, and simple Web-based deployment.

Information Builders and iWay Software Professional Services specialize in building custom-tailored Web decision support systems. We offer service packages designed for quick implementations, and we use the latest technologies to incorporate leading-edge capabilities into our solutions – including a wide range of wireless and mobile options.

What is the need for decision support?

Today, decision making is more difficult. The need for decision-making speed has increased, overload of information is common, and there is more distortion of information. On the positive side, there is a greater emphasis on fact-based decision making. A complex decision-making environment creates a need for computerized decision support. Research and case studies provide evidence that a well-designed and appropriate computerized decision support system can encourage fact-based decisions, improve decision quality, and improve the efficiency and effectiveness of decision processes.

Most managers want more analyses and specific decision-relevant reports quickly. Certainly, we have many and increasing information needs. The goal of DSS is to create and use better information. Today, there is a pressing need to use technology to help make important decisions. Decision makers perform better with the right information at the right time. In general, computerized decision support can help transfer and organize knowledge. Effective decision support provides managers more independence to retrieve and analyze data and documents to obtain facts and results, as they need them.

Introducing more and better decision support in an organization does create changes and challenges for managers. Using a smart phone with decision support applications or a Tablet PC with wireless connectivity to the Internet and corporate databases requires new skills and new knowledge.

What are categories of DSS ?

The organization on the basis of a DSS, often bound to large data warehouses and serve many administrators.

Desktop, single user DSS on the other hand, are generally small systems that can be installed on a single Administrator's computer.

What are the approaches of DSS?

- For a specific purpose, give priority to function or to be able to establish the connection and other facilities as needed to provide the ability to utilize the most.
- A general decision support system that allows maximum package design.

What are the structure and properties ?

Basically decision support systems have three structures that database, model base, and user interface.

The first model-based decision support system has been developed in 1980.

Programmed and Non-programmed Decisions

There are two types of decisions - programmed and non-programmed decisions.

Programmed decisions are basically automated processes, general routine work, where:

- These decisions have been taken several times.

- These decisions follow some guidelines or rules.

For example, selecting a reorder level for inventories, is a programmed decision.

Non-programmed decisions occur in unusual and non-addressed situations, so:

- It would be a new decision.
- There will not be any rules to follow.
- These decisions are made based on the available information.
- These decisions are based on the manger's discretion, instinct, perception and judgment.

For example, investing in a new technology is a non-programmed decision.

Decision support systems generally involve non-programmed decisions. Therefore, there will be no exact report, content, or format for these systems. Reports are generated on the fly.

What are Attributes of a DSS

- Adaptability and flexibility
- High level of Interactivity
- Ease of use
- Efficiency and effectiveness
- Complete control by decision-makers

- Ease of development
- Extendibility
- Support for modeling and analysis
- Support for data access
- Standalone, integrated, and Web-based

Characteristics of a DSS

- Support for decision-makers in semi-structured and unstructured problems.
- Support for managers at various managerial levels, ranging from top executive to line managers.
- Support for individuals and groups. Less structured problems often requires the involvement of several individuals from different departments and organization level.
- Support for interdependent or sequential decisions.
- Support for intelligence, design, choice, and implementation.
- Support for variety of decision processes and styles.
- DSSs are adaptive over time.

What are the benefits of DSS?

- Improves efficiency and speed of decision-making activities.
- Increases the control, competitiveness and capability of futuristic decision-making of the organization.
- Facilitates interpersonal communication.
- Encourages learning or training.
- Since it is mostly used in non-programmed decisions, it reveals new approaches and sets up new evidences for an unusual decision.
- Helps automate managerial processes.

What are classification of DSS?

- There are several ways to classify DSS. Hoi Apple and Whinstone classifies DSS as follows:
- Text Oriented DSS: It contains textually represented information that could have a bearing on decision. It allows documents to be electronically created, revised and viewed as needed.
- Database Oriented DSS: Database plays a major role here; it contains organized and highly structured data.

- Spreadsheet Oriented DSS: It contains information in spread sheets that allows create, view, modify procedural knowledge and also instructs the system to execute self-contained instructions. The most popular tool is Excel and Lotus 1-2-3.
- Solver Oriented DSS: It is based on a solver, which is an algorithm or procedure written for performing certain calculations and particular program type.
- Rules Oriented DSS: It follows certain procedures adopted as rules.
- Rules Oriented DSS: Procedures are adopted in rules oriented DSS. Export system is the example.
- Compound DSS: It is built by using two or more of the five structures explained above.

What is Business Intelligent?

Business intelligence (BI) can be described as "a set of techniques and tools for the acquisition and transformation of raw data into meaningful and useful information for business analysis purposes".[1] The term "data surfacing" is also more often associated with BI functionality. BI technologies are capable of handling large amounts of structured and sometimes unstructured data to help identify, develop and otherwise create new strategic business opportunities. The goal of BI is to allow for the easy interpretation of these large volumes of data. Identifying new opportunities and implementing an effective strategy based on insights can provide businesses with a competitive market advantage and long-term stability.[2]

BI technologies provide historical, current and predictive views of business operations. Common functions of business intelligence technologies are reporting, online analytical processing, analytics, data mining, process mining, complex event processing, business performance management, benchmarking, text mining, predictive analytics and prescriptive analytics.

BI can be used to support a wide range of business decisions ranging from operational to strategic. BI can be also used as an effective tool in entrepreneurial ventures.[3] Basic operating decisions include product positioning or pricing. Strategic business decisions include priorities, goals and directions at the broadest level. In all cases, BI is most effective when it combines data derived from the market in which a company operates (external data) with data from company sources internal to the business such as financial and operations data (internal data). When combined, external and internal data can provide a more complete picture which, in effect, creates an "intelligence" that cannot be derived by any singular set of data.[4] Amongst myriad uses, BI tools empower organisations to gain insight into new markets, assess demand and suitability of products and services for different market segments and gauge the impact of marketing efforts.

What are Differences between BI and DSS ?

DSS :

- Data warehouse is not required.
- Serves to all the events.
- The result of academic research.
- There are aims to find solutions to the problems.

BI :

- Data warehouse is required.
- Serves to the big events.
- By software companies.
- For commercial purposes.
- Includes a working area that is wider than the DSS.

3.1.2. REGRESSION ANALYSIS

In statistical modeling, regression analysis is a statistical process for estimating the relationships among variables. It includes many techniques for modeling and analyzing several variables, when the focus is on the relationship between a dependent variable and one or more independent variables (or 'predictors'). More specifically, regression analysis helps one understand how the typical value of the dependent variable (or 'criterion variable') changes when any one of the independent variables is varied, while the other independent variables are held fixed. Most commonly, regression analysis estimates the conditional expectation of the dependent variable given the independent variables – that is, the average value of the dependent variable when the independent variables are fixed. Less commonly, the focus is on a quantile, or other location parameter of the conditional distribution of the dependent variable given the independent variables. In all cases, the estimation target is a function of the independent variables called the regression function. In regression analysis, it is also of interest to characterize the variation of the dependent variable around the regression function which can be described by a probability distribution. A related but distinct approach is necessary condition analysis(NCA), which estimates the maximum (rather than average) value of the dependent variable for a given value of the independent variable (ceiling line rather than central line) in order to identify what value of the independent variable is necessary but not sufficient for a given value of the dependent variable.

Many techniques for carrying out regression analysis have been developed. Familiar methods such as linear regression and ordinary least squares regression are parametric, in that the regression function is defined in terms of a finite number of unknown parameters that are estimated from the data. Nonparametric regression refers to techniques that allow the regression function to lie in a specified set of functions, which may be infinite-dimensional.

The performance of regression analysis methods in practice depends on the form of the data generating process, and how it relates to the regression approach being used. Since the true form of the data-generating process is generally not known, regression analysis often depends to some extent on making assumptions about this process. These assumptions are sometimes testable if a sufficient quantity of data is available. Regression models for prediction are often useful even when the assumptions are moderately violated, although they may not perform optimally. However, in many applications, especially with small effects or questions of causality based on observational data, regression methods can give misleading results.

Linear regression

In linear regression, the model specification is that the dependent variable, yi is a linear combination of the parameters (but need not be linear in the independent variables). For example, in simple linear regression for modeling n data points there is one independent variable xi, and two parameters, beta 0 and beta 1

straight line:

In multiple linear regression, there are several independent variables or functions of independent variables.

Adding a term in xi2 to the preceding regression gives:

parabola:

Nonlinear regression

When the model function is not linear in the parameters, the sum of squares must be minimized by an iterative procedure. This introduces many complications which are summarized in Differences between linear and non-linear least squares

Power and sample size calculations

There are no generally agreed methods for relating the number of observations versus the number of independent variables in the model. One rule of thumb suggested by Good and

Hardin is N = m ^ {n}, where N is the sample size, n is the number of independent variables and m is the number of observations needed to reach the desired precision if the model had only one independent variable. [27] For example, a researcher is building a linear regression model using a dataset that contains 1000 patients (N). If the researcher decides that five observations are needed to precisely define a straight line (m), then the maximum number of independent variables the model can support is 4, because

Linear Regression Function / Equation How to Calculate

3.2. FIFTH WEEK

3.2.1. ASSOCIATION RULES

Association rule learning is a method for discovering interesting relations between variables in large databases. It is intended to identify strong rules discovered in databases using some measures of interestingness.[1] Based on the concept of strong rules, Rakesh Agrawal et al.[2] introduced association rules for discovering regularities between products in large-scale transaction data recorded by point-of-sale (POS) systems in supermarkets. For example, the rule {\displaystyle \{\mathrm {onions,potatoes} \}\Rightarrow \{\mathrm {burger} \}} \{{\mathrm {onions,potatoes}}\}\Rightarrow \{{\mathrm {burger}}\} found in the sales data of a supermarket would indicate that if a customer buys onions and potatoes together, they are likely to also buy hamburger meat. Such information can be used as the basis for decisions about marketing activities such as, e.g., promotional pricing or product placements. In addition to the above example from market basket analysis association rules are employed today in many application areas including Web usage mining, intrusion detection, Continuous production, and bioinformatics. In contrast with sequence mining, association rule learning typically does not consider the order of items either within a transaction or across transactions.

ALGORITHMS

Apriori algorithm

Apriori [8] uses a breadth-first search strategy to count the support of itemsets and uses a candidate generation function which exploits the downward closure property of support.

Eclat algorithm

Eclat[9] (alt. ECLAT, stands for Equivalence Class Transformation) is a depth-first search algorithm using set intersection. It is a naturally elegant algorithm suitable for both sequential as well as parallel execution with locality enhancing properties. It was first introduced by Zaki, Parthasarathy, Li and Ogihara in a series of papers written in 1997.

Mohammed Javeed Zaki, Srinivasan Parthasarathy, M. Ogihara, Wei Li: New Algorithms for Fast Discovery of Association Rules. KDD 1997.

Mohammed Javeed Zaki, Srinivasan Parthasarathy, Mitsunori Ogihara, Wei Li: Parallel Algorithms for Discovery of Association Rules. Data Min. Knowl. Discov.

FP-growth algorithm

In the first pass, the algorithm counts occurrence of items (attribute-value pairs) in the dataset, and stores them to 'header table'. In the second pass, it builds the FP-tree structure by inserting instances. Items in each instance have to be sorted by descending order of their frequency in the dataset, so that the tree can be processed quickly. Items in each instance that do not meet minimum coverage threshold are discarded. If many instances share most frequent items, FP-tree provides high compression close to tree root.

Recursive processing of this compressed version of main dataset grows large item sets directly, instead of generating candidate items and testing them against the entire database. Growth starts from the bottom of the header table (having longest branches), by finding all instances matching given condition. New tree is created, with counts projected from the original tree corresponding to the set of instances that are conditional on the attribute, with each node getting sum of its children counts. Recursive growth ends when no individual items conditional on the attribute meet minimum support threshold, and processing continues on the remaining header items of the original FP-tree

3.3. SIX WEEK

3.3.1. CLUSTER ANALYSIS

Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters). It is a main task of exploratory data mining, and a common technique for statistical data analysis, used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, bioinformatics, data compression, and computer graphics.

Cluster analysis itself is not one specific algorithm, but the general task to be solved. It can be achieved by various algorithms that differ significantly in their notion of what constitutes a cluster and how to efficiently find them. Popular notions of clusters include groups with small distances among the cluster members, dense areas of the data space, intervals or particular statistical distributions. Clustering can therefore be formulated as a multi-objective optimization problem. The appropriate clustering algorithm and parameter settings (including values such as the distance function to use, a density threshold or the number of expected clusters) depend on the individual data set and intended use of the results. Cluster analysis as such is not an automatic task, but an iterative process of knowledge discovery or interactive multi-objective optimization that involves trial and failure. It is often necessary to modify data preprocessing and model parameters until the result achieves the desired properties.

Besides the term clustering, there are a number of terms with similar meanings, including automatic classification, numerical taxonomy, botryology (from Greek βότρυς "grape") and typological analysis. The subtle differences are often in the usage of the results: while in data mining, the resulting groups are the matter of interest, in automatic classification the resulting discriminative power is of interest.

Cluster analysis was originated in anthropology by Driver and Kroeber in 1932 and introduced to psychology by Zubin in 1938 and Robert Tryon in 1939[1][2] and famously used by Cattell beginning in 1943[3] for trait theory classification in personality psychology.

ALGORITHM

Clustering algorithms can be categorized based on their cluster model, as listed above. The following overview will only list the most prominent examples of clustering algorithms, as there are possibly over 100 published clustering algorithms. Not all provide models for their clusters and can thus not easily be categorized. An overview of algorithms explained in Wikipedia can be found in the list of statistics algorithms.

There is no objectively "correct" clustering algorithm, but as it was noted, "clustering is in the eye of the beholder."[4] The most appropriate clustering algorithm for a particular problem often needs to be chosen experimentally, unless there is a mathematical reason to prefer one cluster model over another. It should be noted that an algorithm that is designed for one kind of model has no chance on a data set that contains a radically different kind of model.[4] For example, k-means cannot find non-convex clusters.

Connectivity-based clustering (hierarchical clustering)

Connectivity based clustering, also known as hierarchical clustering, is based on the core idea of objects being more related to nearby objects than to objects farther away. These algorithms connect "objects" to form "clusters" based on their distance. A cluster can be described largely by the maximum distance needed to connect parts of the cluster. At different distances, different clusters will form, which can be represented using a dendrogram, which explains where the common name "hierarchical clustering" comes from: these algorithms do not provide a single partitioning of the data set, but instead provide an extensive hierarchy of clusters that merge with each other at certain distances. In a dendrogram, the y-axis marks the distance at which the clusters merge, while the objects are placed along the x-axis such that the clusters don't mix.

Connectivity based clustering is a whole family of methods that differ by the way distances are computed. Apart from the usual choice of distance functions, the user also needs to decide on the linkage criterion (since a cluster consists of multiple objects, there are multiple candidates to compute the distance to) to use. Popular choices are known as single-linkage clustering (the minimum of object distances), complete linkage clustering (the maximum of object distances) or UPGMA ("Unweighted Pair Group Method with Arithmetic Mean", also known as average linkage clustering). Furthermore, hierarchical clustering can be agglomerative (starting with single elements and aggregating them into clusters) or divisive (starting with the complete data set and dividing it into partitions).

These methods will not produce a unique partitioning of the data set, but a hierarchy from which the user still needs to choose appropriate clusters. They are not very robust towards outliers, which will either show up as additional clusters or even cause other clusters to merge (known as "chaining phenomenon", in particular with single-linkage clustering). In the general case, the complexity is {\displaystyle {\mathcal {O}}(n^{3})} {\mathcal {O}}(n^{3}) for agglomerative clustering and {\displaystyle {\mathcal {O}}(2^{n-1})} {\mathcal {O}}(2^{n-1}) for divisive clustering,[5] which makes them too slow for large data sets. For some special cases, optimal efficient methods (of complexity {\displaystyle {\mathcal {O}}(n^{2})} {\mathcal {O}}(n^{2})) are known: SLINK[6] for single-linkage and CLINK[7] for complete-linkage clustering. In the data mining community these methods are recognized as a theoretical foundation of cluster analysis, but often considered obsolete[citation needed]. They did however provide inspiration for many later methods such as density based clustering.

Centroid-based clustering

In centroid-based clustering, clusters are represented by a central vector, which may not necessarily be a member of the data set. When the number of clusters is fixed to k, k-means clustering gives a formal definition as an optimization problem: find the {\displaystyle k} k cluster centers and assign the objects to the nearest cluster center, such that the squared distances from the cluster are minimized.

The optimization problem itself is known to be NP-hard, and thus the common approach is to search only for approximate solutions. A particularly well known approximative method is Lloyd's algorithm,[8] often actually referred to as "k-means algorithm". It does however only find a local optimum, and is commonly run multiple times with different random initializations. Variations of k-means often include such optimizations as choosing the best of multiple runs, but also restricting the centroids to members of the data set (k-medoids), choosing medians (k-medians clustering), choosing the initial centers less randomly (K-means++) or allowing a fuzzy cluster assignment (Fuzzy c-means).

Most k-means-type algorithms require the number of clusters - {\displaystyle k} k - to be specified in advance, which is considered to be one of the biggest drawbacks of these algorithms. Furthermore, the algorithms prefer clusters of approximately similar size, as they will always assign an object to the nearest centroid. This often leads to incorrectly cut borders in between of clusters (which is not surprising, as the algorithm optimized cluster centers, not cluster borders).

K-means has a number of interesting theoretical properties. First, it partitions the data space into a structure known as a Voronoi diagram. Second, it is conceptually close to nearest neighbor classification, and as such is popular in machine learning. Third, it can be seen as a variation of model based classification, and Lloyd's algorithm as a variation of the Expectation-maximization algorithm for this model discussed below.

Distribution-based clustering

The clustering model most closely related to statistics is based on distribution models. Clusters can then easily be defined as objects belonging most likely to the same distribution. A convenient property of this approach is that this closely resembles the way artificial data sets are generated: by sampling random objects from a distribution.

While the theoretical foundation of these methods is excellent, they suffer from one key problem known as overfitting, unless constraints are put on the model complexity. A more complex model will usually be able to explain the data better, which makes choosing the appropriate model complexity inherently difficult.

One prominent method is known as Gaussian mixture models (using the expectation-maximization algorithm). Here, the data set is usually modelled with a fixed (to avoid overfitting) number of Gaussian distributions that are initialized randomly and whose parameters are iteratively optimized to fit better to the data set. This will converge to a local optimum, so multiple runs may produce different results. In order to obtain a hard clustering, objects are often then assigned to the Gaussian distribution they most likely belong to; for soft clusterings, this is not necessary.

Distribution-based clustering produces complex models for clusters that can capture correlation and dependence between attributes. However, these algorithms put an extra burden on the user: for many real data sets, there may be no concisely defined mathematical model (e.g. assuming Gaussian distributions is a rather strong assumption on the data).

Density-based clustering

In density-based clustering,[9] clusters are defined as areas of higher density than the remainder of the data set. Objects in these sparse areas - that are required to separate clusters - are usually considered to be noise and border points.

The most popular[10] density based clustering method is DBSCAN.[11] In contrast to many newer methods, it features a well-defined cluster model called "density-reachability". Similar to linkage based clustering, it is based on connecting points within certain distance thresholds. However, it only connects points that satisfy a density criterion, in the original variant defined as a minimum number of other objects within this radius. A cluster consists of all density-connected objects (which can form a cluster of an arbitrary shape, in contrast to many other methods) plus all objects that are within these objects' range. Another interesting property of DBSCAN is that its complexity is fairly low - it requires a linear number of range queries on the database - and that it will discover essentially the same results (it is deterministic for core and noise points, but not for border points) in each run, therefore there is no need to run it multiple times. OPTICS[12] is a generalization of DBSCAN that removes the need to choose an appropriate value for the range parameter {\displaystyle \varepsilon } \varepsilon , and produces a hierarchical result related to that of linkage clustering. DeLi-Clu,[13] Density-Link-Clustering combines ideas from single-linkage clustering and OPTICS, eliminating the {\displaystyle \varepsilon } \varepsilon parameter entirely and offering performance improvements over OPTICS by using an R-tree index.

RECENT DEVELOPMENT

In recent years considerable effort has been put into improving the performance of existing algorithms.[15][16] Among them are CLARANS (Ng and Han, 1994),[17] and BIRCH (Zhang et al., 1996).[18] With the recent need to process larger and larger data sets (also known as big data), the willingness to trade semantic meaning of the generated clusters for performance has been increasing. This led to the development of pre-clustering methods such as canopy clustering, which can process huge data sets efficiently, but the resulting "clusters" are merely a rough pre-partitioning of the data set to then analyze the partitions with existing slower methods such as k-means clustering. Various other approaches to clustering have been tried such as seed based clustering.[19]

For high-dimensional data, many of the existing methods fail due to the curse of dimensionality, which renders particular distance functions problematic in high-dimensional spaces. This led to new clustering algorithms for high-dimensional data that focus on subspace clustering (where only some attributes are used, and cluster models include the relevant attributes for the cluster) and correlation clustering that also looks for arbitrary rotated ("correlated") subspace clusters that can be modeled by giving a correlation of their attributes.[20] Examples for such clustering algorithms are CLIQUE[21] and SUBCLU.[22]

Ideas from density-based clustering methods (in particular the DBSCAN/OPTICS family of algorithms) have been adopted to subspace clustering (HiSC,[23] hierarchical subspace clustering and DiSH[24]) and correlation clustering (HiCO,[25] hierarchical correlation clustering, 4C[26] using "correlation connectivity" and ERiC[27] exploring hierarchical density-based correlation clusters).

Several different clustering systems based on mutual information have been proposed. One is Marina Meilă's variation of information metric;[28] another provides hierarchical clustering.[29] Using genetic algorithms, a wide range of different fit-functions can be optimized, including mutual information.[30] Also message passing algorithms, a recent development in Computer Science and Statistical Physics, has led to the creation of new types of clustering algorithms.

3.3.2. DATA AND RULES

What is the output of a decision support system?

Traditionally, one would answer the output of a DSS is "relevant information". Correct, but DSS designers and builders need to understand exactly what information a manager needs and wants in a specific decision situation. So more exactly, the output of a computerized decision support system may be quantitative results from models, analyses and displays of historical data, displays of facts in various formats, recommendations, and retrieved relevant documents. DSS outputs are the direct result of the interaction of user inputs, stored or accessed data and documents and analytical and retrieval processes in the computerized system. One hopes DSS outputs decrease uncertainty in a decision situation and positively impact decisions!

Outputs describe the information that comes from the programmed process in the DSS. An output may be a map, a chart, a tabular data summary, a printed report or a data file. DSS outputs include forms, objects and other representations for inputs and manipulation by a user and representations that display results from queries, analyses and rules for users. Decision support information may be data points on a chart, text, stored or computer generated images and even sounds. Decision support system output may be descriptive or prescriptive. Any DSS output must be "suitable for human interpretation". DSS outputs should inform the user of the system about the decision situation that is supported.

Managers and their support staffs need to consider what information and analyses are actually needed to support management and business decisions. Some managers need both detailed transaction data and summarized data. Most managers only want summarized data. Managers usually want lots of charts and graphs; a few only want tables of numbers. Many managers want information provided routinely or periodically and some want information available on-line and on demand. Managers want financial analyses and some managers want primarily "soft", non-financial or qualitative information.

The main tasks of DSS is verification, formatting, storing, distribution and analyzing of data considering the needs of the specific user. The user needs can be various and the system should provide answers to users in an automatic or manual way depending on a queries status - standard or non-standard. The standard queries are periodic activity reports like: list of protected species or range of protection activities. An automatic process of other nonstandard queries would be conducted depending on the needs of a user. However, it should be pointed out that most nonstandard queries will be realized manually by using a tool such as ArcGIS Desktop. Actually, the DSS was already used for realizing several queries described in Figure 1 (frame output).

What is the data of a decision support system?

We can produce the data ourselves or we can take a certain database and analyze it. Firstly we should decide a subject for the DSS, than we should find a database or create a database.

3.4. SEVEN WEEK

3.4.1. REGRESSION ANALYSIS APPLICATON

First we installed matlab on the computer. We preferred matlab, but can be made excel. We have chosen a simple example and applied it. We have worked on the temperature data.

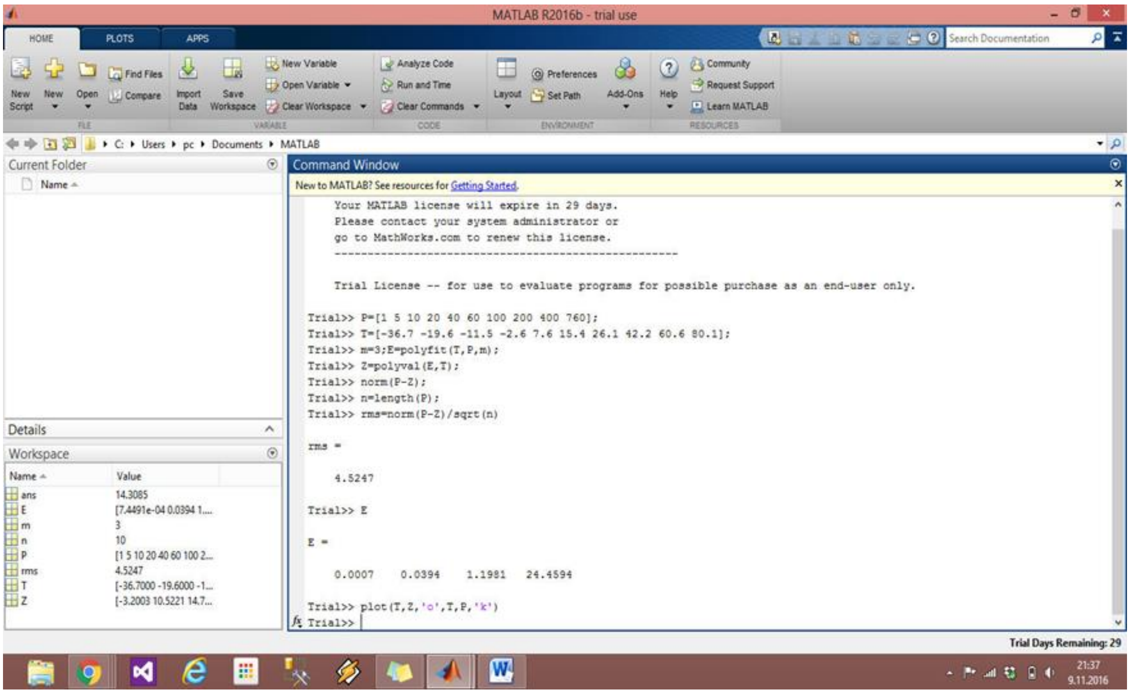

Figure 1:Matlab

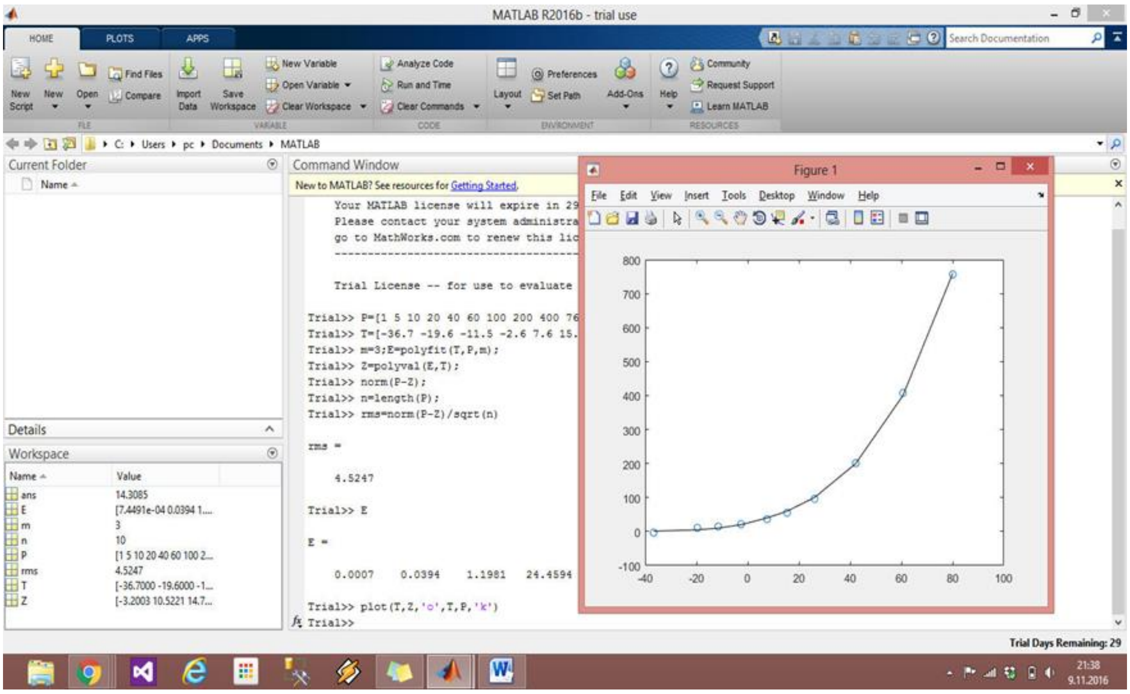

Figure 2: Regression analysis and graph drawing

We have already calculated the variation in the data and have drawn the graph in matlab.

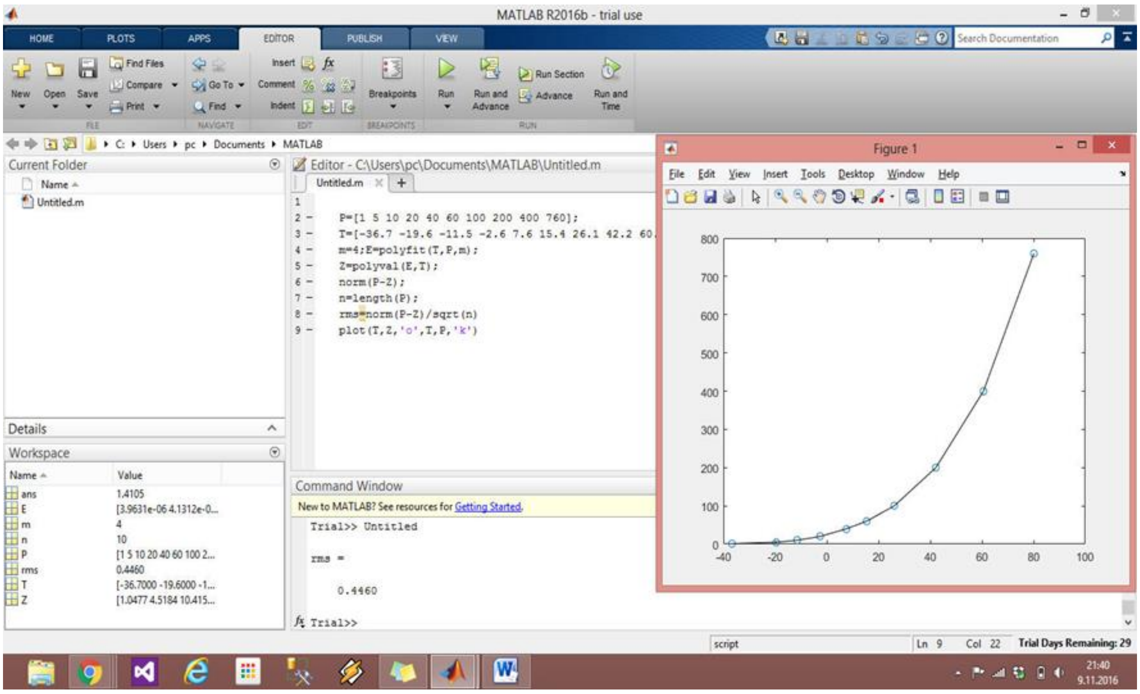

Figure 3: Regression analysis and graph drawing

We changed the coefficient of the data and again we were drawn a chart.

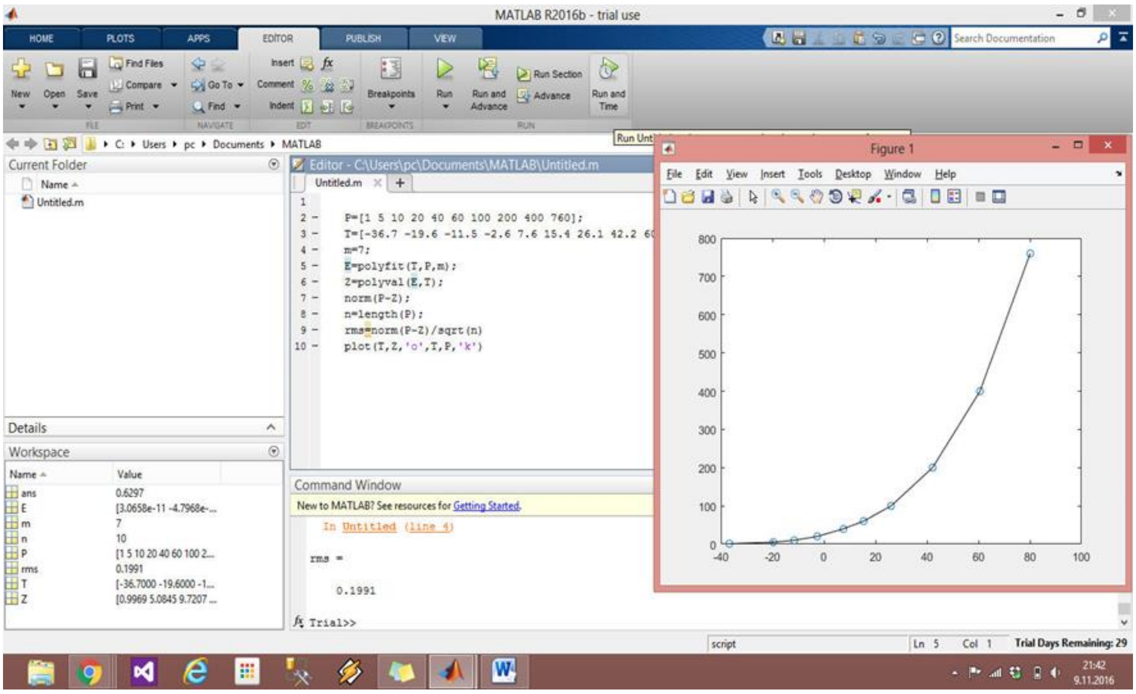

Figure 4: Regression analysis and graph drawing

3.5. EIGHT WEEK

3.5.1. CLUSTERING ANALYSIS APPLICATON

First we installed matlab on the computer. We preferred matlab, but can be made excel. We have chosen a simple example and applied it. We have worked on the temperature data.

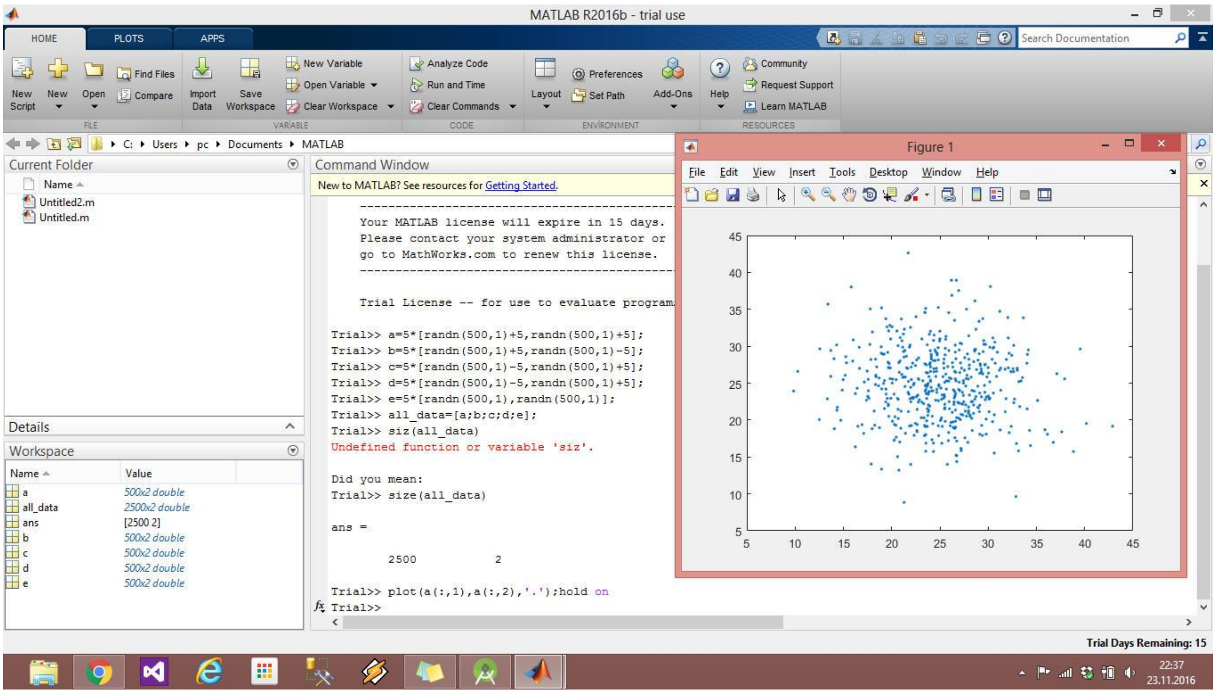

Figure 1:Matlab

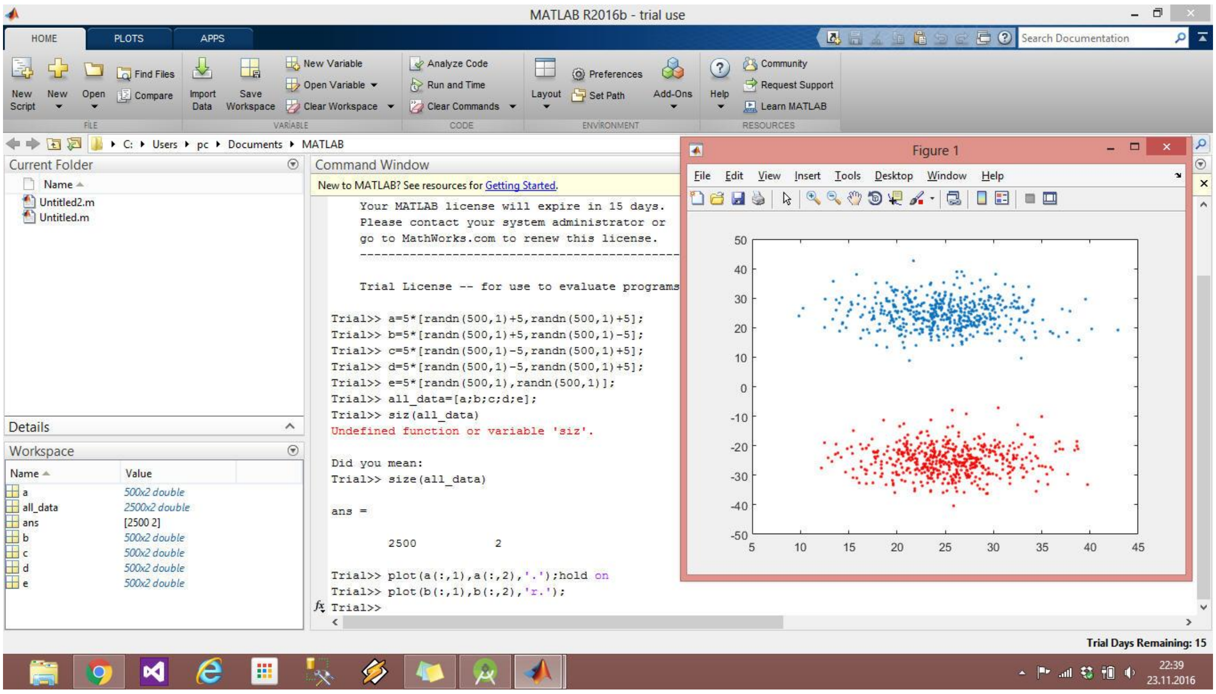

Figure 2:Clustering analysis and graph drawing

We have already calculated the variation in the data and have drawn the graph in matlab.

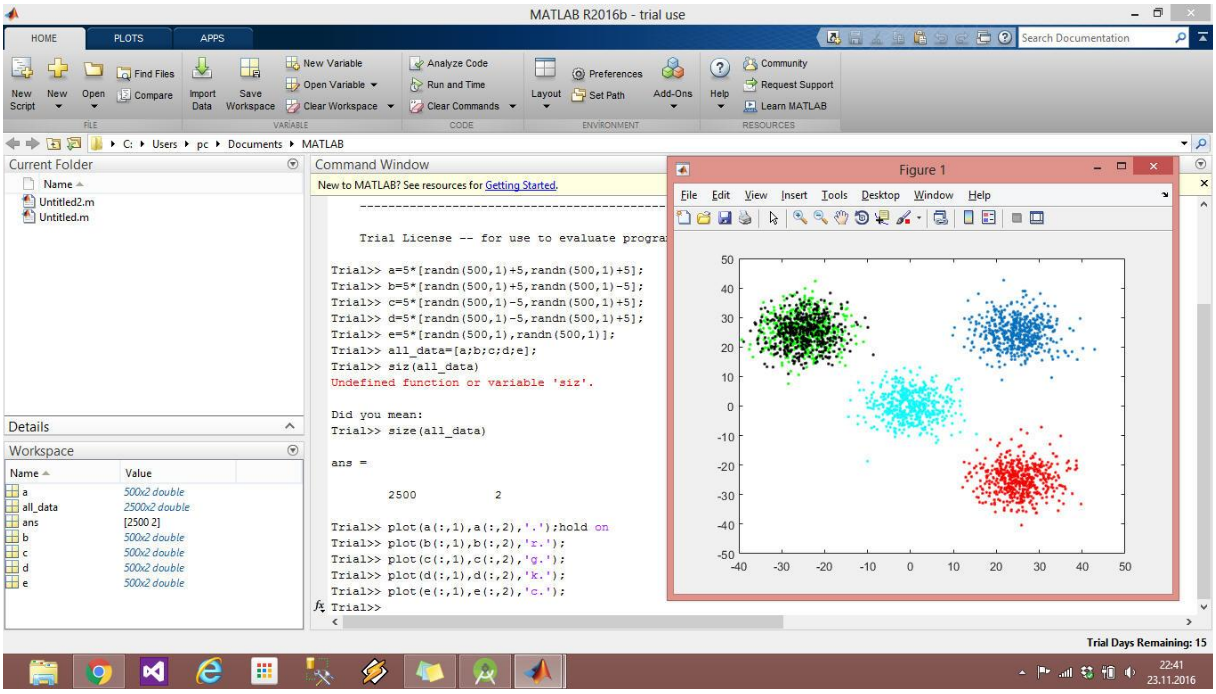

Figure 3: Clustering analysis and graph drawing

We changed the coefficient of the data and again we were drawn a chart.

3.6. NINETH WEEK

3.6.1. WEB SERVICES EXAMPLE

We researched android web service. We have found some examples and examined them. An example of active android studio. We can easily create a restful web service application in android to authenticate or save information into the external database such as oracle, mysql, postgre sql, sql server using other application developed in java, .net, php etc languages. That is what we are going to do.

Before developing web services application, you must have basic knowledge of SOAP and Restful web services. That is why, we are going to discuss basic points about web services such as what is web service and brief information about SOAP and Restful web services.

A web service is a standard for exchanging information between different types of applications irrespective of language and platform. For example, an android application can interact with java or .net application using web services.

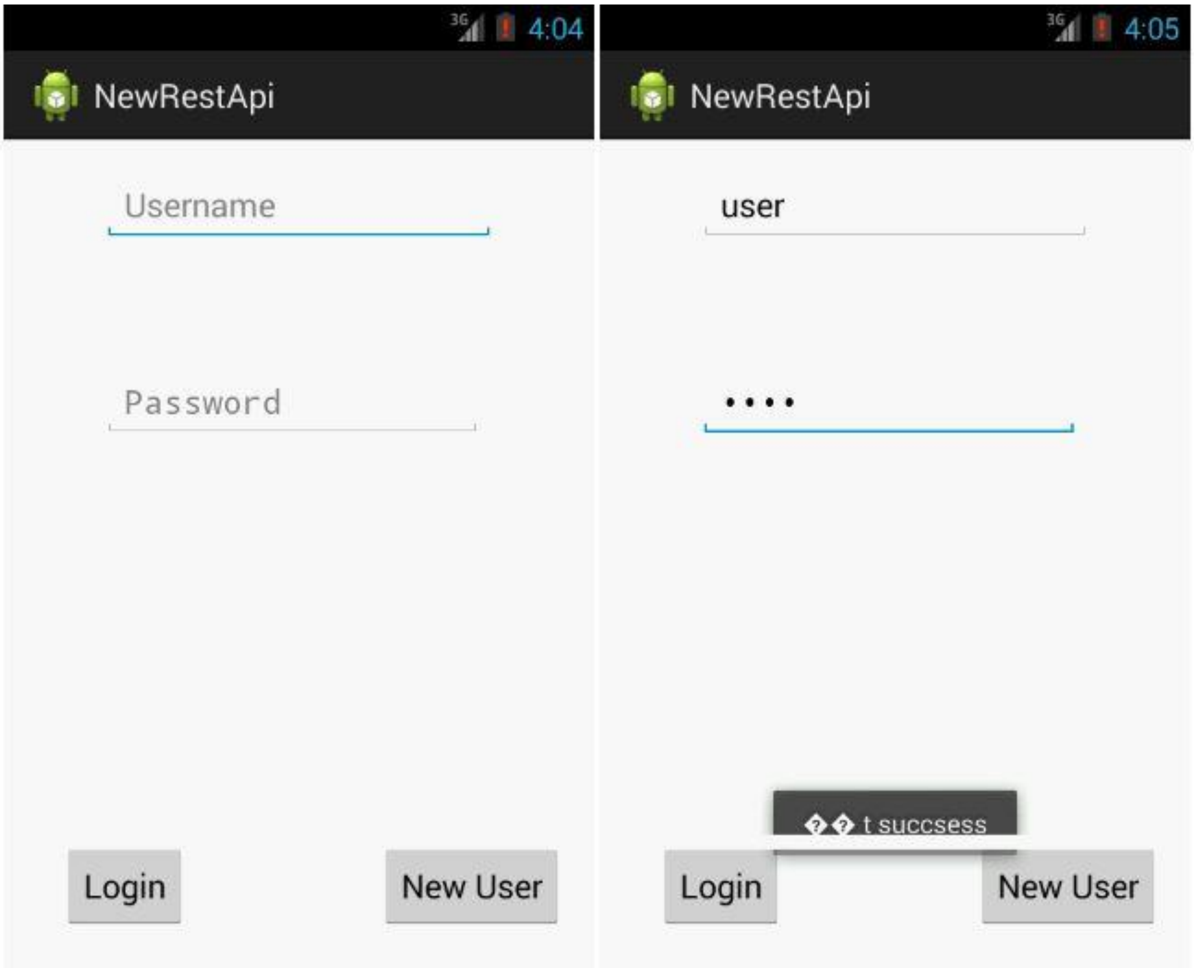

Figure3.1: Web service example

As seen in Figure 3.1, we are trying to make web service android. We tried to connect to oracle database. We have many mistakes. We tried to solve them.

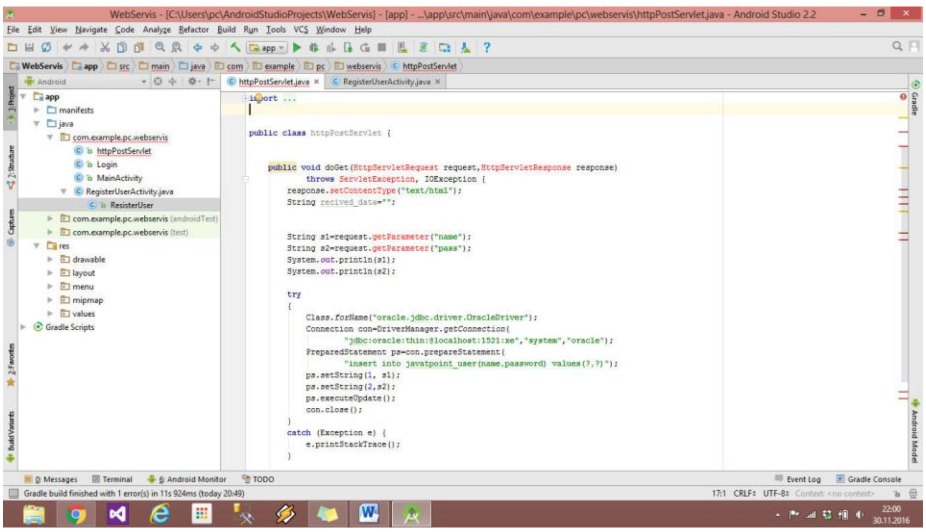

Figure 3.2: Android web service code

4. TENTH WEEK

4.1. .NET WEB SERVICE EXAMPLE

We used Visual Studio. We are writing asp.net. We used SQL Server 2012 for database connections.

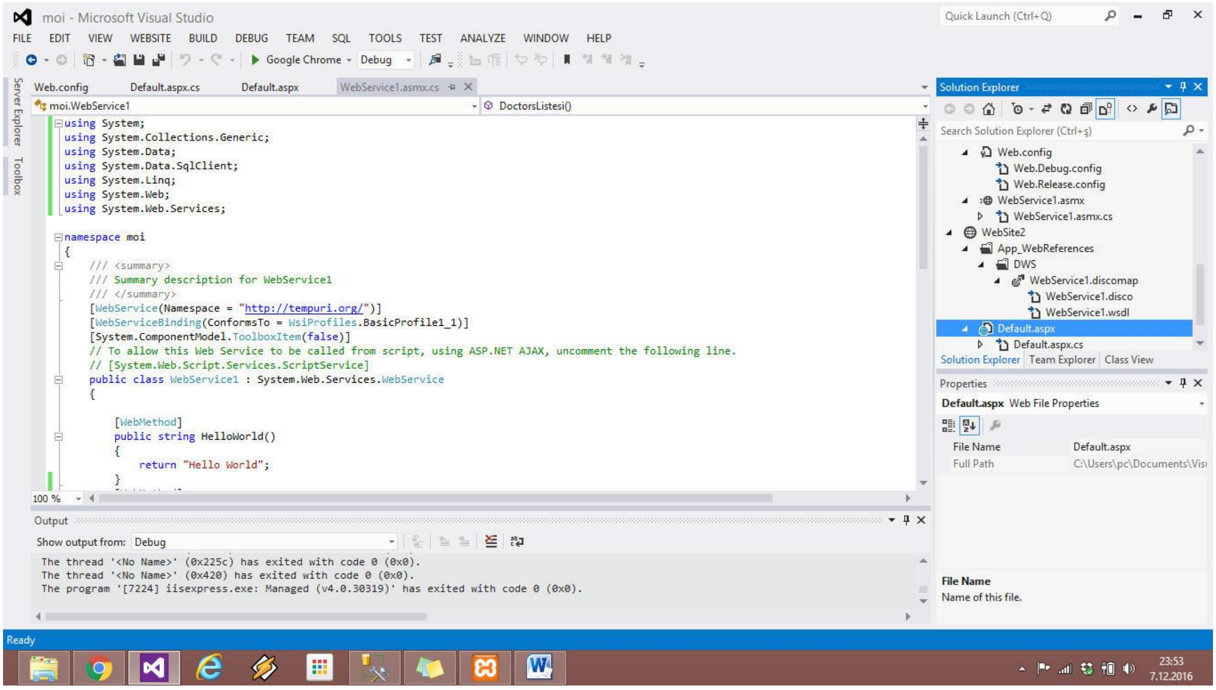

Figure 4.1. Web Service in Visual Studio

5. ELEVEN & TWELFTH WEEK

At the following week, we continued to work on webservice from last week example. We tried to apply our project. But we did not do that. Than we decided to use another webservice function.

6. THIRTEENTH WEEK

In this week, We worked on connection between matlab and android studio with webservice. But we could not do it yet.

7. FOURTEENTH WEEK

In last week, we have completed our last changes. We have changed the Web service method in android studio. We used ksop library for call to web service. We wrote a web service for every function. As show in *Figure 7.1* we have a working mechanism. For example; we have milk data created by more than one kind of animal according to years. Let’s say we picked an animal from animal varieties. When we chose an animal, we call a web service according to chosen animal from “.Net”. This web service does the regression analysis from data of come from database. Result of web service is a json object. We did parse the object in android studio. Than we showed on Table Layout or we showed optionally on graphic.

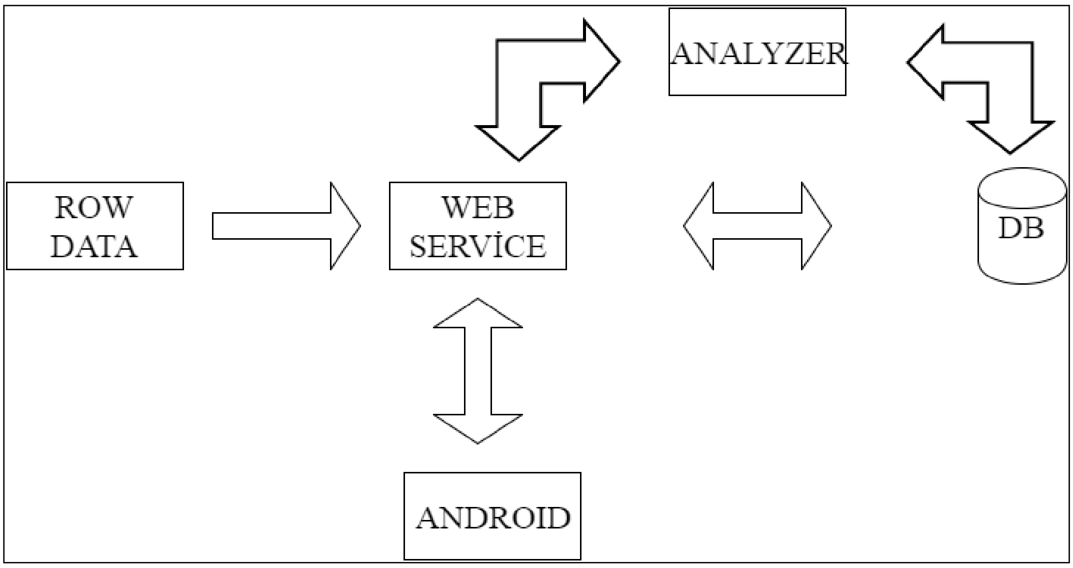

Figure 7.1 Operational Mechanism

As show in *figure 7.2*, this figure represents our homepage. User can choose a process or function from inside of spinner.

figure 7.2: Homepage

As show in *Figure 7.3*, When user choose a type of animal, this figure comes to screen.

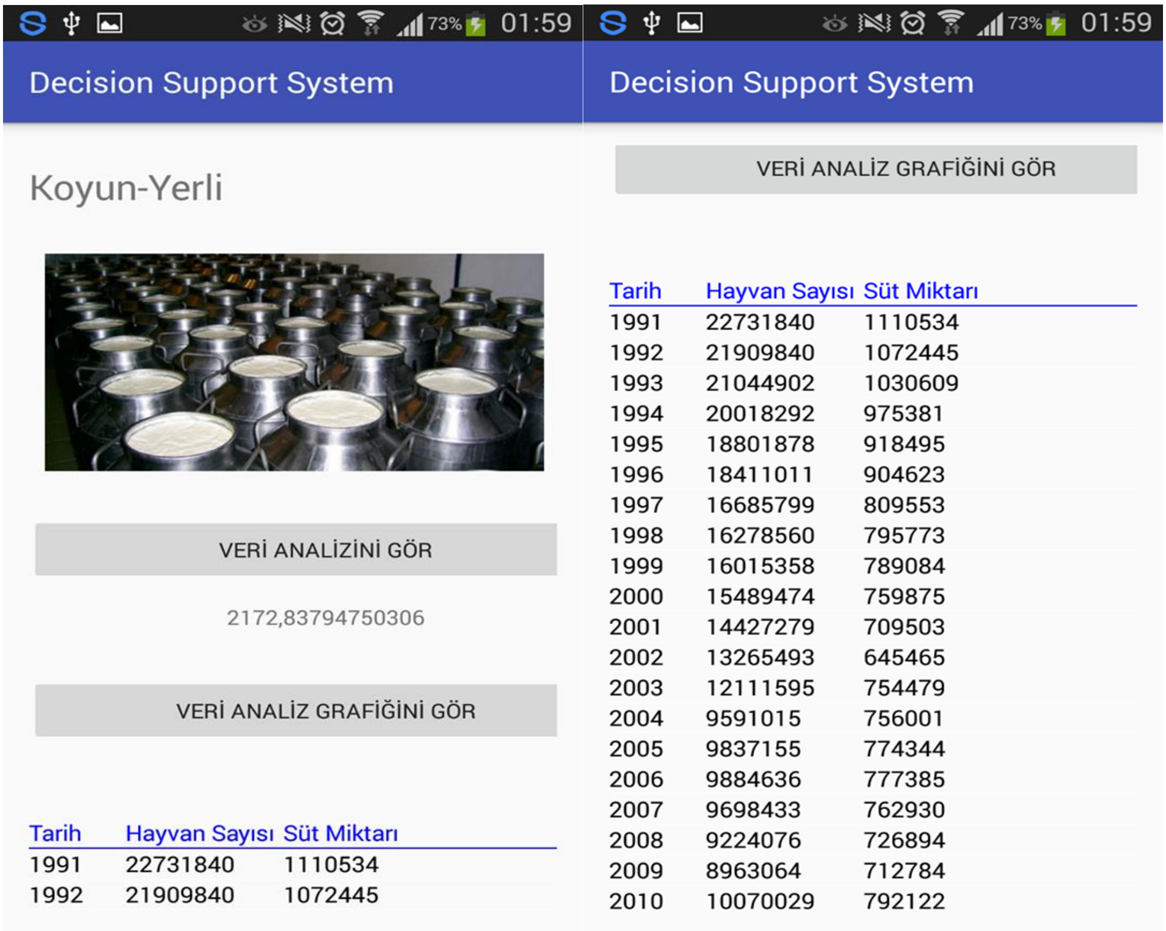

Figure 7.3 Analyze of choosen type of animal

As show in ***Figure 7.4***, When user push the “veri analizi grafiğini göster button” from show in ***Figure 7.3***.

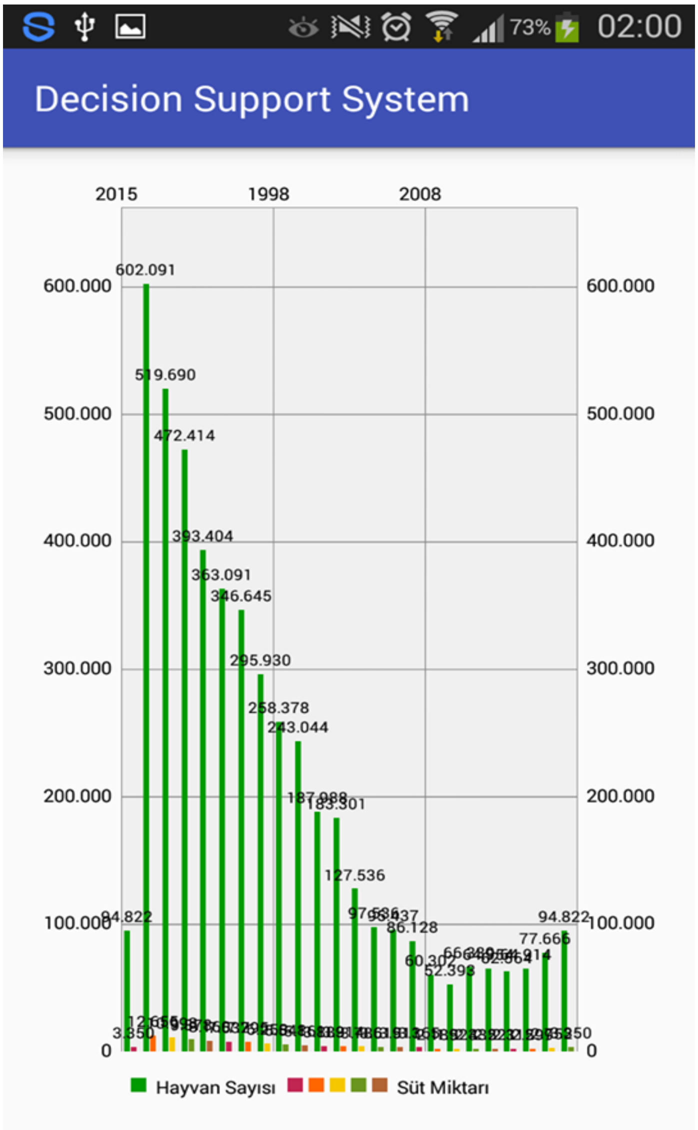

Figure 7.4 Graphic of Analyze

Conclusionly, we finished our project without any problems.

8. PROBLEMS ENCOUNTERED

We have many encountered problems. We have Http Servlet Request and cannot import some libraries. We are working on these. System or Android Studio didn’t recognize R class. We didn’t solve this problem. We got error on web.service.aspx in tenth week. We solved this error with create a new project that include empty web applications.

9. CHANGES IN REQUIREMENTS

We didn’t changes in requırements this week.

10.OVERALL ASSESMENT OF THE PROJECT

We are learned web service. We learned to connection between android and .net. We learned to use android layouts. We learned to working with planning.

11.REFERENCES

http://www.androidhive.info/2012/01/android-login-and-registration-with-php-mysql-and-sqlite/

http://www.javatpoint.com/android-web-service

http://www.buraksenyurt.com/post/Basit-Bir-Web-Service-Uygulamasc4b1-bsenyurt-com-dan.aspx

http://www.truiton.com/2015/04/android-chart-example-mp-android-chart-library/

http://stackoverflow.com/questions/27191610/display-sqlite-table-data-row-by-row-in-android-using-tablelayout-dynamically

http://stackoverflow.com/questions/3957094/convert-hashmap-tostring-back-to-hashmap-in-java

http://stackoverflow.com/questions/10164741/get-jsonarray-without-array-name

http://stackoverflow.com/questions/16143562/string-comparison-android

http://androidexample.com/Restful_Webservice_Call_And_Get_And_Parse_JSON_Data-_Android_Example/index.php?view=article_discription&aid=101

http://stackoverflow.com/questions/17398019/how-to-convert-datatable-to-json-in-c-sharp

https://weblog.west-wind.com/posts/2008/Sep/03/DataTable-JSON-Serialization-in-JSONNET-and-JavaScriptSerializer

http://accord-framework.net/docs/html/T_Accord_Statistics_Models_Regression_Linear_SimpleLinearRegression.htm

https://tr.wikipedia.org/wiki/Regresyon_analizi