An International Journal
JSCDSS
http://www.jscdss.com
Vol.5 No.4 August 2018: 26-32
Article history:
Accepted 12 August 2018
Published online 13 August 2018

Journal of Soft Computing and Decision Support Systems

E-ISSN: 2289-8603

Prediction of Patients' Mortality during Hospitalizations

Mojtaba Zare [a,*], Janusz Wojtusiak [a], Mehrbakhsh Nilashi [b]

[a] Department of Health Administration and Policy, College of Health and Human Services, George Mason University, Fairfax, VA 22030-4444, USA
[b] Faculty of Computing, Universiti Teknologi Malaysia, 81310, Skudai, Johor, Malaysia

*** Corresponding authors email addresses**: mzare@gmu.edu

Abstract

In this study, we predict patients' mortality in the session that a patient is hospitalized. We focus on both lab results and vital signs collected in the first 24 hours of a patient admission to predict his/her mortality within that hospitalization. We use MIMIC-III dataset for data analysis and building predictive models. We include only patients with age of 18 or above resulting in a sample of 38,578 patients. Independent variables include patients' demographic information, lab results and vital signs. The dependent variable is whether the patient dies within that hospitalization. We use Weka 3.8 for data analysis and model building. After randomly splitting the data into 80 and 20 percent, then we use the 80 percent of the data for feature selection as well as training the prediction models. As a result, we construct four prediction models using Bayes Network, Logistic Regression, Naïve Bayes and Random Forest. After constructing these models, we test them on the remaining 20 percent of the data. We use Receiver Operating Characteristic (ROC) area, precreation and recall values to compare the accuracy of these models with each other. Result showed that the highest ROC area belongs to the Bayes Network at 0.824 followed closely by Logistic Regression at 0.817 for the test set.

Keywords: Mortality, MIMIC-III, Classification algorithm, Prediction models

1. Introduction

Mortality prediction of hospitalized patients is a crucial task. It is important to accurately predict the mortality of a patient because it benefits both the patient and health care resources (Awad et al., 2017; Lee et al., 2015). Over the past recent years, machine learning models have been proposed to predict mortality of patients with different type of diseases such as breast and lung cancer (Martín-Sánchez et al., 2016; Malvezzi et al., 2015), and chronic kidney disease (Haapio et al., 2017; Vijayarani et al., 2015). However, most of previous studies have focused on building data mining models to predict patient morality after patients' ICU admission or after discharge (Awad et al., 2017; Wojtusiak et al., 2017). Hence, early mortality prediction regardless of patients' admission to ICU or patients discharge remains an open challenge. In this study, we predict patients' mortality in the session that a patient is hospitalized. We focus on both lab results and vital signs collected in the first 24 hours of a patient admission to predict his/her mortality within that hospitalization.

2. Method

2.1 Dataset

In this study, we used MIMIC-III (Medical Information Mart for Intensive Care III) dataset for data analysis and building prediction models. MIMIC-III is a large freely database containing information of 46,520 patients who stayed in critical care units of the Beth Israel Deaconess Medical Center between 2001 and 2012. The database contains patients' information such as "demographics, vital sign measurements made at the bedside, laboratory test results, procedures, medications, caregiver notes, imaging reports, and mortality (both in and out of hospital)" (Johnson et al., 2016).

2.2 Variables

In this study, we included only patients with age of 18 or above. This sample consists of 38,578 patients. Because patients may have more than one admission, we randomly chose one admission for each patient. After conducting an extensive examination of data variables, we included patients' demographic information, lab results and vital signs as independent variables. Patient demographic information consisted of age, race and gender. We focused on both lab results and vital signs variables collected in the first 24 hours of a patient admission to predict his/her mortality within that hospitalization. Lab result included information about ALBUMIN, BILIRUBIN, CREATININE, CHLORIDE, GLUCOSE, HEMATOCRIT, HEMOGLOBIN, PLATELET, POTASSIUM, SODIUM, BUN, WBC and WBC_count rate of the patients. Vital

signs include maximum, minimum and average information about heartrate, sysbp (systolic blood pressure), diasbp (diastolic blood pressure), meanbp (mean blood pressure), resprate (respiration rate (rate of breathing)), tempc (temperature), spo2 (blood oxygen saturation), glucose. In summary, a total of 40 independent variables were selected consisting of 3 demographic variables, 13 lab results and 24 vital signs. The dependent variable was whether a patient dies within that hospitalization (0 or 1).

2.3 Data pre-processing

In this study, data pre-processing has mainly four steps:

i. Creating a table that take one random admission for each patient from "admissions table" in MIMIC-III databse. After that, we will add gender, date of birth (dob) from "patients table" to this table.
ii. Creating a table from "labevents table" which contains lab result of patients within 24 hours after admission.
iii. Creating a table that has vital result of patients within 24 hours after their admissions.
iv. Left join tables created in Step i and Step iii to the table created in Step i.

Step A. First, we randomly chose an admission for each patient from admissions table. In order to do that, we created a column of random number and then based on this column we chose a random admission for each patient. The SQL programming code is presented in Appendix A.
Step B. In this step, we create a table from labevents table and this new table will have lab result of a patient within 24 hours after his/her admission. Lab results include: ALBUMIN, BILIRUBIN, CREATININE, CHLORIDE, GLUCOSE, HEMATOCRIT, HEMOGLOBIN, PLATELET, POTASSIUM, SODIUM, BUN, WBC_count, WBC. The programming code is presented in Appendix B.
Step C. In this step, we created a table that has vital result of a patient within 24 hours after his/her admissions. We used minimum, maximum, average values for heartrate, sysbp (systolic blood pressure), diasbp (diastolic blood pressure), meanbp, resprate (respiration rate), tempc (temperature), spo2 (oxygen saturation), glucose. The programming code is presented in Appendix C.
Step D. In this step, we joined the tables that we created in Step B and Step C to the table created in Step A. The programming code is presented in Appendix D.

2.4. Feature selection

In general, feature selection methods are used to remove irrelevant and redundant attributes that do not play a main role in the accuracy of the predictive model or even sometimes they can decrease the accuracy of the prediction model (Guyon & Elisseeff, 2003). As a result of feature selection, not only we may observe improvement in the accuracy of the prediction model, but also, we will have a less complex model resulting in faster and more cost-effective model. In this study we used "CorrelationAttributeEval" feature selection method in Weka (with threshold set to 0.01). This method "evaluates the worth of an attribute by measuring the correlation (Pearson's) between it and the class" (Weka). Figures in Appendix F belong to the initial dataset before applying feature selection method.

In order to apply the feature selection method, we first randomized the data and then we split it to 80 and 20 percent. The SQL programming code is presented in Appendix E.

Then, we applied the feature selection method to the 80 percent of the data. As a result, 4 features were dropped leaving us with 36 independent variables (see the figure in Appendix G).

2.5 Data analysis

We used Weka 3.8 for data analysis and model building. After applying feature selection to the 80 percent of the data, we used that 80 percent of the data to train the predictive models. We constructed four predictive models using algorithms such as Bayes Network, Logistic Regression, Naïve Bayes, Random Forest. After constructing these models, then we test them on the remaining 20 percent of the data. We used ROC area, precreation and recall values that to compare the accuracy of these models with each other.

3. Results

Table 1 shows the result of ROC area, precision and recall for each model. As we can see the highest ROC area belongs to the Bayes Network at 0.824 followed closely by Logistic Regression at 0.817 for the test set. Random tree overfitted the data and it performed poorly on the test set with the lowest ROC area at 0.712.

Table 2 shows interesting results. The ROC area, precision and recall values in table 2 are for the initial data prior to applying the attribute selection method (all 40 independent features). As we can see, almost all models performed poorly compared to the models trained after applying the feature selection method to the dataset. These echoes the benefits of attribute selection methods that we discussed in Section 2.4.

JSCDS
E-ISSN: 2289-8603

Table 1
ROC, Precision and Recall for predictive models

	Training set			Test set		
Model	ROC area	Precision	Recall	ROC area	Precision	Recall
Bayes Network	0.826	0.426	0.580	0.824	0.412	0.558
Logistic Regression	0.816	0.688	0.236	0.817	0.633	0.225
Naïve Bayes	0.804	0.439	0.463	0.805	0.434	0.463
Random tree	1.000	1.000	0.983	0.712	0.304	0.245

Table 2
ROC, Precision and Recall for predictive models (no attribute selection; 80 percent training, 20 percent test set)

	Test set		
Model	**ROC area**	**Precision**	**Recall**
Bayes Network	0.814	0.408	0.553
Logistic regression	0.809	0.698	0.235
Naïve Bayes	0.801	0.320	0.625
Random tree	0.699	0.288	0.220

4. Future work

There is a potential research that can be conducted in future to construct a regression model to predict the number of days until a patient dies from his/her admission date for patients who predicted to die. By doing that, physicians and healthcare professionals will have a better understanding about condition of patients and as a result more effective measures can be taken accordingly.

5. Conclusion

There are many data mining systems in hospitals to predict patient mortality. However, these models are mainly predicting patient mortality after 1 or 2 days of patient ICU admission or after patient discharge. In this study, we proposed predictive model, in particular Bayes Network that will predict patient mortality only 24 hours after his/her admission. The proposed framework can be used in hospitals to help physicians and healthcare professionals in decision making and taking right measurements which as a result can directly benefits patients and health care resources.

Appendix A:

```
-- 1-creating a random column between
100 and 10^10
create temp table mytemp as
(select subject_id ,ethnicity,
hadm_id, admittime,
hospital_expire_flag, random() *
10000000000000000 + 100 as rndm
from admissions)
-- SELECT 58976

-- 2-selecting a random row for each
patient
create temp table test1 as
(select subject_id,  min (rndm) as
min_rndm
from mytemp
group by subject_id)
-- SELECT 46520

-- 3-creating rand_admissions
select m.subject_id ,ethnicity,
hadm_id, admittime,
hospital_expire_flag
into rand_admissions
from mytemp m inner join test1 t
on m.rndm=min_rndm
-- SELECT 46520

-- number of patient who died in rand
table
select count (*) from rand_admissions
where hospital_expire_flag=1
-- 5001

-- 4-adding dob and gender to
rand_admissions
select r.subject_id, hadm_id,
p.gender, p.dob, admittime,
ethnicity, hospital_expire_flag
into rand_admissions_gender_dob
from
rand_admissions r left join patients
p
on r.subject_id=p.subject_id
-- SELECT 46520

--
select * from
rand_admissions_gender_dob
limit 10

-- 5-calculating age on the date of
admission (greater than 18)
select * ,  DATE_PART('year',
admittime ) - DATE_PART('year', dob)
as age
into rand_admissions_gender_dob_18
from rand_admissions_gender_dob
where  DATE_PART('year', admittime )
- DATE_PART('year', dob) >= 18
-- SELECT 38578
```


JSCDS E-ISSN: 2289-8603

```
-- 6-changing  AGE above 89  TO 90.
Update rand_admissions_gender_dob_18
SET age=90
where age >89
-- UPDATE 1991

-- 7- grouping race. In this step, we
grouped all the races to three
groups, white, black, others.
SELECT *,
CASE
      WHEN lower(ethnicity) like
'%white%' THEN 'white'
      WHEN lower(ethnicity) like
'%black%' THEN 'black'
      ELSE 'others'
END AS race_group
into rand_admissions_gender_dob_18_s2
FROM rand_admissions_gender_dob_18;
-- SELECT 38578

select * from
rand_admissions_gender_dob_18_s2
limit 10 ;

--7.2 droping ethnicity, dob,
admittime
ALTER TABLE
rand_admissions_gender_dob_18_s2
DROP COLUMN admittime;


select * from
rand_admissions_gender_dob_18_s2
limit 10
```

Appendix B:

```
-- 1- we want to get the information
of lab events within 24 hours of an
admission.
-- first, we add admittime to the
labevents table and then we substract
the admittime from charttime.
Substraction should be between 0 and
24, because we are only interested to
have the patient lab information
within 24 hours of admission.
-- In all of these steps we right
join to the rand_admissions, because
we want to keep all the patients that
we have in rand_admissions table.

select  r.hadm_id , itemid,  valuenum
, valueuom
into lab_result_within24h
from labevents l right join
rand_admissions r
on l.hadm_id = r.hadm_id
where (DATE_PART ('day',charttime-
admittime)*24 + DATE_PART
('hours',charttime-admittime))<='24'
and (DATE_PART ('day',charttime-
admittime)*24 + DATE_PART
('hours',charttime-admittime)) >='0'
;
-- SELECT 3522418

--
select count (distinct hadm_id) from
lab_result_within24h
--  44506 -- some admission doesn't
have lab info within 24 hours after
admission (the where condition
above). That's why this number
(44506) is less than 46520 which is
the number of rows for
rand_admissions table.


-- 2- now, we will pivot table
lab_result_within24h in a way that
one row for one hadm_id in
rand_admissions.

select hadm_id,
      avg(CASE WHEN itemid=50862
THEN valuenum ELSE null END) as
ALBUMIN,
      avg(CASE WHEN itemid=50885
THEN valuenum ELSE null END) as
BILIRUBIN,
      avg(CASE WHEN itemid=50912
THEN valuenum ELSE null END) as
CREATININE,
      avg(CASE WHEN itemid=50902
THEN valuenum ELSE null END) as
CHLORIDE,
      avg(CASE WHEN itemid=50931
THEN valuenum ELSE null END) as
GLUCOSE,
      avg(CASE WHEN itemid=51221
THEN valuenum ELSE null END) as
HEMATOCRIT,
      avg(CASE WHEN itemid=51222
THEN valuenum ELSE null END) as
HEMOGLOBIN,
      avg(CASE WHEN itemid=51265
THEN valuenum ELSE null END) as
PLATELET,
      avg(CASE WHEN itemid=50971
THEN valuenum ELSE null END) as
POTASSIUM,
      avg(CASE WHEN itemid=50983
THEN valuenum ELSE null END) as
SODIUM,
```


JSCDS E-ISSN: 2289-8603

```
      avg(CASE WHEN itemid=51006
THEN valuenum ELSE null END) as BUN,
      avg(CASE WHEN itemid=51300
THEN valuenum ELSE null END) as
WBC_count,
      avg(CASE WHEN itemid=51301
THEN valuenum ELSE null END) as WBC
into lab_result_within24h_pivot
from lab_result_within24h
group by hadm_id
-- SELECT 44506
```

Appendix C:

```
select
hadm_id, min (heartrate_min)
heartrate_min,
max(heartrate_max)heartrate_max
,avg(heartrate_mean) heartrate_mean,min
(sysbp_min) sysbp_min,max
(sysbp_max)sysbp_max,avg (sysbp_mean)
sysbp_mean,min (diasbp_min) diasbp_min ,
max (diasbp_max) diasbp_max,avg
(diasbp_mean) diasbp_mean,min
(meanbp_min) meanbp_min,max (meanbp_max)
meanbp_max,avg (meanbp_mean)
meanbp_mean,min (resprate_min)
resprate_min,max (resprate_max)
resprate_max,
avg (resprate_mean) resprate_mean,min
(tempc_min) tempc_min,max (tempc_max)
tempc_max,
avg (tempc_mean) tempc_mean,min
(spo2_min) spo2_min,max (spo2_max)
spo2_max,avg (spo2_mean) spo2_mean,min
(glucose_min) glucose_min,max
(glucose_max) glucose_max,avg
(glucose_mean) glucose_mean
into vitalsfirstday2_unique_hadmid
from  vitalsfirstday2
group by hadm_id
-- SELECT 56215
```

Appendix D:

```
-- 1) join lab_result_within24h_pivot
(table in step B) with
rand_admissions_gender_dob_18_s2
(table in step A)

select r.* ,
ALBUMIN,BILIRUBIN,CREATININE,CHLORIDE
,GLUCOSE,HEMATOCRIT,HEMOGLOBIN,PLATEL
ET,POTASSIUM,SODIUM,BUN,WBC_count,WBC
into addedup_s1
from
rand_admissions_gender_dob_18_s2 r
left join lab_result_within24h_pivot
l
on
r.hadm_id=l.hadm_id
-- SELECT 38578

-- join addedup_s1 with
vitalsfirstday2_unique_hadmid (table
in step C)
select a.* , heartrate_min
,heartrate_max  ,heartrate_mean
,sysbp_min ,sysbp_max ,sysbp_mean
,diasbp_min ,diasbp_max ,diasbp_mean
,meanbp_min,meanbp_max ,meanbp_mean
,resprate_min
,resprate_max,resprate_mean
,tempc_min,tempc_max,tempc_mean,spo2_
min,spo2_max,spo2_mean,glucose_min,gl
ucose_max,glucose_mean
into addedup_s2
from addedup_s1 a left join
vitalsfirstday2_unique_hadmid v
on
a.hadm_id=v.hadm_id
-- SELECT 38578
```

Appendix E:

```
  -- feature selection.
-- 1- adding a  column with random number
to addedup_s2
select *, random() * 1000000000000000 + 100
as rndm
into addedup_s2_rand
from addedup_s2
-- SELECT 38578

-- 2- top 80 percent for  feature selection
and training . (38578 * 0.8 = 30862.4 )
select *
into training
from addedup_s2_rand
order by rndm desc
limit 30862
-- SELECT 30862

 -- 3- the rest would be 38578-30862=7716
select *
into test
from addedup_s2_rand
order by rndm asc
limit 7716


select count (*) from test
where hospital_expire_flag=1
-- 972
```

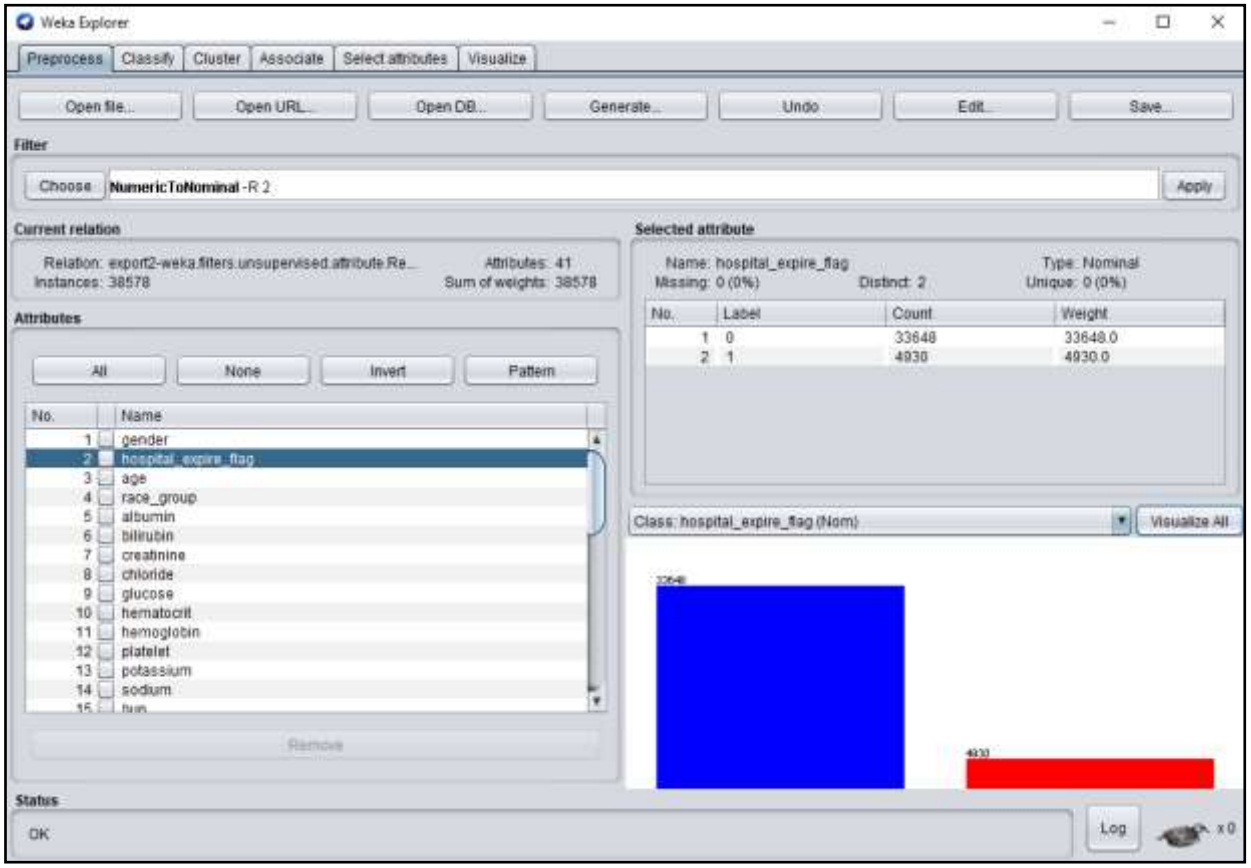

E-ISSN: 2289-8603

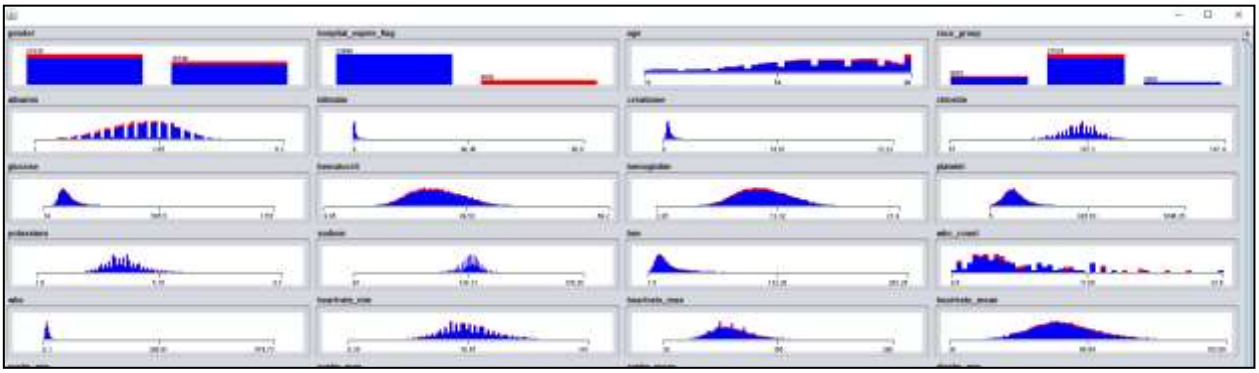
```
-- 4- dropping rndm column                    DROP COLUMN rndm;

ALTER TABLE training
DROP COLUMN rndm;

ALTER TABLE test
```

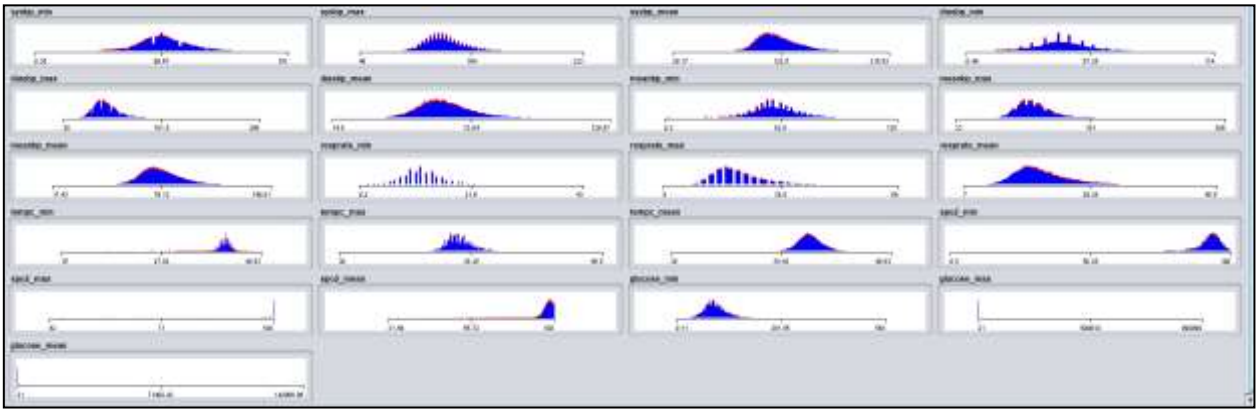
Appendix F:



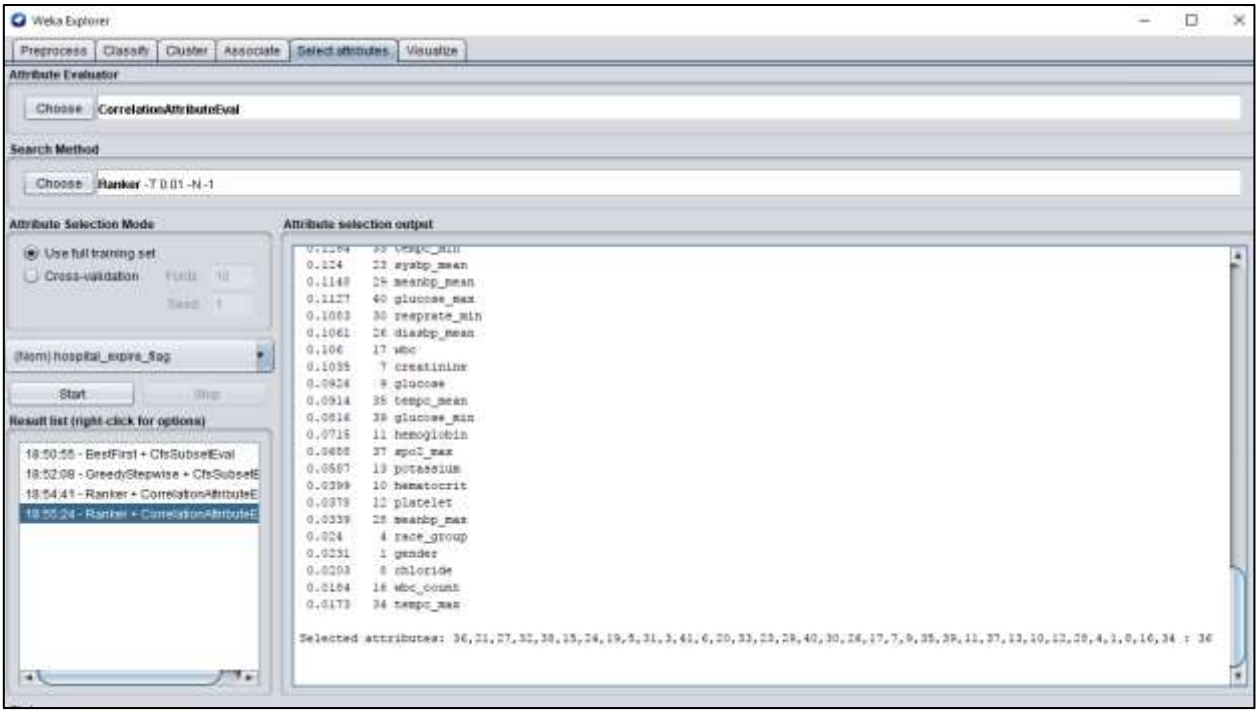


E-ISSN: 2289-8603

Appendix F (Cont.):



Appendix G:



References

Awad, A., Bader-El-Den, M., McNicholas, J., & Briggs, J. (2017). Early hospital mortality prediction of intensive care unit patients using an ensemble learning approach. International Journal of Medical Informatics, 108, 185-195.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. Journal of machine learning research, 3(Mar), 1157-1182.

Haapio, M., Helve, J., Grönhagen-Riska, C., & Finne, P. (2017). One-and 2-Year Mortality Prediction for Patients Starting Chronic Dialysis. Kidney International Reports, 2(6), 1176-1185.

Johnson, A. E., Pollard, T. J., Shen, L., Lehman, L. W. H., Feng, M., Ghassemi, M., ... & Mark, R. G. (2016). MIMIC-III, a freely accessible critical care database. Scientific data, 3.

Lee, J., Maslove, D. M., & Dubin, J. A. (2015). Personalized mortality prediction driven by electronic medical data and a patient similarity metric. PloS one, 10(5), e0127428.

Malvezzi, M., Bertuccio, P., Rosso, T., Rota, M., Levi, F., La Vecchia, C., & Negri, E. (2015). European cancer mortality predictions for the year 2015: does lung cancer have the highest death rate in EU women?. Annals of Oncology, 26(4), 779-786.

Martín-Sánchez, J. C., Clèries, R., Lidón, C., González-de Paz, L., Lunet, N., & Martínez-Sánchez, J. M. (2016). Bayesian prediction of lung and breast cancer mortality among women in Spain (2014–2020). Cancer epidemiology, 43, 22-29.

Vijayarani, S., Dhayanand, S., & Phil, M. (2015). Kidney disease prediction using SVM and ANN algorithms. International Journal of Computing and Business Research (IJCBR), 6(2).

Weka software. Online from: http://weka.sourceforge.net/doc.dev/weka/attributeSelection/CorrelationAttributeEval.html

Wojtusiak, J., Elashkar, E., & Nia, R. M. (2017, February). C-Lace: Computational Model to Predict 30-Day Post-Hospitalization Mortality. In HEALTHINF (pp. 169-177).

JSCDS
E-ISSN: 2289-8603