

R/Shiny를 활용한 금융 조기경보시스템 개발

이용주*, 김영민**

Development of Financial Early Warning System with R/Shiny

YongJu Lee*, Young Min Kim**

본 논문은 교육부 및 한국연구재단의 4단계 BK21 사업(경북대학교 컴퓨터학부 지능융합 소프트웨어 교육연구단)으로 지원된 연구임(4120240214871). 이 논문은 2016년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2016R1D1A1B02008553).

요약

은행 파산과 같은 금융 위기가 자주 발생된다면 국가적으로 큰 손실이 될 수 있으므로 사전에 이런 조짐을 분석하고 위기에 적절히 대응할 수 있는 조기경보시스템(EWS, Early Warning System)의 개발은 매우 중요하다. 본 연구에서는 EWS 구현을 위해 워크플로우 전체 과정을 자동화하고, R/Shiny와 MySQL 등 오픈 소스 플랫폼을 사용하여 대화형 EWS 웹 애플리케이션과 대쉬보드 가시화를 구현한다. 특히, R/Shiny는 기존의 웹 언어인 HTML, CSS, 또는 JavaScript 등을 사용할 필요 없이 대규모 비즈니스 웹 애플리케이션 구축을 매우 쉽고 강력하게 만들어 줄 수 있다. 우리는 전문가 면담과 같은 질적 평가 방식을 채택하여 제안된 시스템과 기존의 Java 기반 시스템을 비교하여 성능 분석을 수행한다. 분석 결과 제안된 시스템이 효율성과 만족도 측면에서 보다 더 우수함을 보인다.

Abstract

If financial crises such as bankruptcy occur frequently, it could be a huge loss for the country. Hence it is very important to develop an early warning system(EWS) that can analyze such signs in advance and respond appropriately to crises. In this study, the entire workflow process is automated to implement EWS, and an interactive EWS Web application and dashboard visualization are implemented using open source platforms such as R/Shiny and MySQL. In particular, R/Shiny can make it very easy and powerful to build large-scale business Web applications without the need to use existing Web languages such as HTML, CSS, or JavaScript. We adopt qualitative evaluation methods such as expert interviews to perform performance analysis by comparing the proposed system with existing Java-based systems. The analysis results show that the proposed system is superior in terms of efficiency and satisfaction.

Keywords

R program, shiny, early warning system, dashboard visualization, workflow automation

* 경북대학교 IT대학 컴퓨터학부 교수(교신저자)
- ORCID: <https://orcid.org/0000-0002-1705-4967>
** 경북대학교 자연과학대학 통계학과 부교수
- ORCID: <https://orcid.org/0000-0001-6295-1758>

• Received: Mar. 09, 2024, Revised: May 13, 2024, Accepted: May 16, 2024
• Corresponding Author: YongJu Lee
School of Computer Science and Engineering, Kyungpook National University, 80, Daehak-ro, Buk-gu, Daegu, Korea
Tel.: +82-53-550-7285, Email: yongju@knu.ac.kr

I. 서론

2008년 리만 브라더스(Lehman Brothers)와 2023년 실리콘밸리 은행(SVB)의 파산, 퍼스트 리퍼블릭 은행(FRB)뱅크런 사태, 그리고 근래 한국의 새마을금고 부실과 같이 최근 금융산업 위기가 총체적으로 증폭되어 국가 경제에 하나의 큰 불안 요소가 되고 있다[1]. 이와 같은 금융 위기가 자주 발생된다면 국가적으로 막대한 손실이 초래될 수 있으므로 사전에 이런 조짐을 분석하고 위기에 대응할 수 있는 금융 조기경보시스템(EWS, Early Warning System)의 중요성이 점차 증가되고 있다. EWS란 금융사의 재무자료 및 거시경제 변수와 통계 기법을 활용하여 금융사의 장래 부실화 가능성을 예측하는 계량 시스템이라고 할 수 있다[2].

EWS는 향후 재무상태가 나빠질 가능성이 높은 금융사 및 재무상태 악화를 초래할 위험 요소를 조기에 판별하는데 그 목적이 있으며, 신용평가 분석 과정이 대부분 통계적 기법의 활용으로 평가되기 때문에 담당자의 주관적인 판단이 최소화되어 사전 예방을 효율적으로 수행할 수 있다. EWS는 모형에 사용되는 데이터의 가용성(Availability), 정확성(Accuracy), 그리고 적합성(Suitability)에 따라 그 품질이 크게 좌우된다. 따라서 데이터를 획득할 수 없거나, 데이터의 정확도가 떨어지거나, 또는 데이터가 그 상황에 적합하게 맞지 않을 경우, EWS의 예측 품질은 저하될 수밖에 없다. 따라서 효율적인 신용평가 작업을 위해 전체 업무 수행 사이클의 자동화가 요구되고 데이터베이스 구축의 선진화 및 지능화 작업이 필요하다.

본 연구에서는 EWS 구현을 위해 데이터 정의, 데이터 처리, 데이터 분석 등의 전체 과정을 자동화하고, 오픈 소스 플랫폼인 MySQL DBMS와 R/Shiny 패키지를 사용하여 웹 서버 환경으로 대시보드 가시화(Dashboard visualization)를 구현한다. 현재 금융 EWS의 워크플로우 자동화 및 시각화를 위해 대화형 웹 애플리케이션으로 확장 발전시키는 것은 이 분야의 최대 관심사 중 하나이다. 이를 위해 워크플로우 기반 실행 엔진을 개발하고, 빅데이터 및 통계 분석에 가장 널리 사용되는 R 프로그램[3]을 사용

하여 세련되고 직관적인 대시보드를 구현한다. Shiny 패키지[4]는 R 코드를 HTML(Hyper Text Markup Language), CSS(Cascading Style Sheets), 또는 JavaScript 코드로 변환함으로써 대화형 웹 애플리케이션 구축을 상당히 쉽게 만들어 준다. 따라서 본 연구는 R/Shiny를 활용하여 대화형 EWS 웹 애플리케이션을 개발해 보고 이로부터 습득된 노하우(Know-how)와 이슈들을 소개한다. 본 논문의 구성은 2장에서 관련 연구를 살펴보고, 3장에서 금융 조기경보시스템 개발에 대해 기술하고, 4장에서 성능을 분석한다. 그리고 5장에서 결론을 내린다.

II. 관련 연구

2.1 조기경보 모델

조기경보 모델이란 금융 시스템 및 참여 기관을 정기적으로 모니터링하는 알고리즘의 집합을 의미한다[2]. 이 모델을 사용하여 개별 금융기관 및 시스템 자체의 건전성 수준을 효과적으로 정량화하고, 이 정량화를 기반으로 은행 및 기타 금융사에 영향을 미치는 잠재적 문제 및 위험과 위기를 감지한다. EWS 모델은 자본비율변동 모델과 부실회사분류 모형으로 크게 구분이 되고, 식 (1)과 (2) 같은 형식을 취한다[5].

$$y_{t+1} = g(x_t) + \varepsilon_{t+1} \quad (1)$$

$$z_{t+1} \text{ i.i.d. Bernoulli}(p(x_t)) \quad (2)$$

여기서, t 는 현재 시간이고 $t+1$ 은 기본 상태가 예측되는 미래 시간이라고 가정한다. y_{t+1} 은 미래 시간 $t+1$ 에서 회사의 재무상태를 나타내는 결과 변수이고, x_t 는 t 까지의 시간에 수집된 재무상태 집합이다. $g(x_t)$ 와 $p(x_t)$ 는 EWS 모델 또는 알고리즘을 나타내며, ε_{t+1} 은 시간 $t+1$ 에서의 예측 오류이다. 단, $p(x_t)$ 는 확률을 나타내며, *i.i.d.*(independently and identically distributed)는 서로 독립이고 동일한 분포를 의미하며, *Bernoulli*($p(x_t)$)는 $p(x_t)$ 를 확률값으로 갖는 베르누이 분포이다.

식 (2)에서 z_{t+1} 은 0과 1의 값을 가지게 된다 ($z_{t+1}=1$ 은 비정상회사, $z_{t+1}=0$ 은 정상회사를 의미한다). y_t 값은 기업의 재무상태와 밀접한 관련이 있으므로 그에 따라 움직이는 적절한 결과 변수를 선택해야 하는데, y_t 는 다양한 변수가 가능하다(예를 들면, 금융회사의 자본비율). x_t 값은 위험과 위기가 발생하기 전에 y_{t+1} 의 변동을 설명하고 예측할 수 있는 일련의 주요 재무상태가 될 수 있는데, 이는 국가마다 다른 특징들을 사용하므로 그 국가 현지의 조건 고려가 중요하다. 예를 들면, 개별 은행의 RoA(Return on Asset), RoE(Return on Equity), 부동산 지수, 수익률 곡선 변화, 은행 소비자 예금 변화, GDP(Gross Domestic Product), CDS(Credit Default Swap) 스프레드, 환율 변화 등이 될 수 있다.

식 (1)과 (2)의 알고리즘을 위해서는 다양한 모델(예, 로지스틱 회귀, Random Forest, XGBoost, 자본비율변동)을 사용할 수 있지만 각 모델마다 장단점이 있을 수 있으므로 데이터의 가용성, 정확성, 그리고 적합성 등을 고려하여 선택할 필요가 있다[5]. 본 논문에서는 이러한 모델 중 가장 널리 사용되고 있는 자본비율변동(Change in equity) 모델을 설명한다. 자본비율변동 모델은 정상회사와 비정상회사를 분류하지 않는 접근법이므로 모든 자료를 활용할 수 있어 모델의 성능을 높일 수 있다. 이는 회귀(Regression) 모델을 기반으로 각 금융기관에 대한 미래 RoA를 예측하여 기본 확률을 생성한다.

자본비율변동 모델에서 기본 기준은 시간 $t+1$ 에서 자기자본/자산 비율이 어떤 임계값보다 작다는 것이다. 즉, 임계값을 예를 들어 4%로 설정하면 조건(또는 이벤트)이 식 (3)을 만족하면 회사가 불이행 상태에 있음을 의미한다.

$$R_{t+1} = \frac{E_{t+1}}{A_{t+1}} < 4\% \quad (3)$$

결과적으로 기본 확률은 $\Pr[R_{t+1} < 4\%]$ 로 표현된다. 하지만 R_{t+1} 값을 직접 예측하는 것은 어렵기 때문에 식 (4)와 같이 근사값으로 계산할 수 있다.

$$\begin{aligned} R_{t+1} &= \frac{E_{t+1}}{A_{t+1}} \approx \frac{E_{t+1}}{A_t} = \frac{E_t + (E_{t+1} - E_t)}{A_t} \quad (4) \\ &= R_t + RoA_{t+1} \end{aligned}$$

마지막 식에서 R_t 값은 시간 t 에서 알려져 있으므로 유일한 불확실성은 RoA_{t+1} 이 된다. 기본 임계값이 4%로 설정된 경우 기본 확률은 식 (5)와 같다.

$$\begin{aligned} \Pr[R_{t+1} < 4\%] &\approx \Pr[R_t + RoA_{t+1} < 4\%] \quad (5) \\ &= \Pr[RoA_{t+1} < 4\% - R_t] \end{aligned}$$

이제 $y_{t+1} = RoA_{t+1}$ 로 설정하여 식 (1)을 사용하면 2단계의 기본 확률은 식 (6)과 같다.

$$\begin{aligned} \Pr[RoA_{t+1} < 4\% - R_t] &= \Pr[g(x_t) + \varepsilon_{t+1} < 4\% - R_t] \\ &= \Pr[\varepsilon_{t+1} < 4\% - R_t - g(x_t)] \quad (6) \end{aligned}$$

여기서 R_t 및 $g(x_t)$ 는 시간 t 에서 알려진 상수이고, $g(x_t)$ 는 일반적으로 선형 회귀(Linear regression)가 된다. 즉 식 (7)과 같이 된다.

$$g(x_t) = \beta'x_t = \beta_0 + \beta_1x_{1,t} + \dots + \beta_px_{p,t} \quad (7)$$

마지막 단계에서 예측 오차 ε_{t+1} 의 분포를 지정하는데, 이를 지정하는 방법에는 여러 가지가 있을 수 있다. 즉, 과거 경험적 관찰로 얻은 값과 모델로 추정된 값의 차를 사용하거나, 데이터로부터 추정된 파라메트릭 분포(예, 정규 분포)를 사용하거나(그림 1 참조), 또는 향후 경제 상황을 반영한 주관적인 분포를 고려할 수 있다. 자본비율변동 모형의 결과값을 예를 들어 설명해 보면, $R_t = 5\%$ 이고 $g(x_t) = 1\%$ 라고 가정하면 ε_{t+1} 의 분포를 고려할 때 금융사의 부실확률 $\Pr[R_{t+1} < 4\%] = \Pr[\varepsilon_{t+1} < -2\%] = 0.025(2.5\%)$ 가 된다. 따라서 그림 1에서 -2% 보다 작은 2.5% 가 부실 금융사로 예측된다.

한편, EWS 자동화 및 대쉬보드 가시화 관련 연구로 FSS-EWS[6]는 2008년부터 현재까지 다양한 모형들을 개발해 오고 있다. 하지만 본 논문과 같은 워크플로우 자동화나 반응형 Shiny Server 애플리케이션은 아직 구현되어 있지 않다.

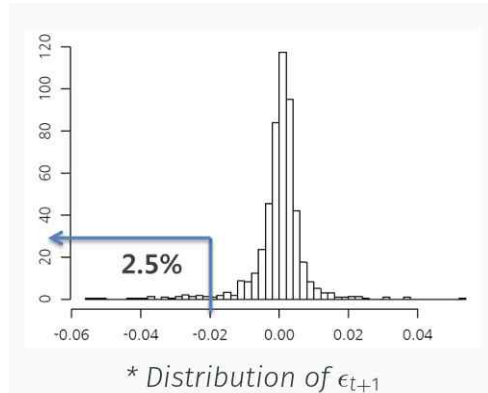


그림 1. 금융사의 부실 확률
Fig. 1. Insolvent propability of financial company

2.2 R과 Shiny

R 프로그램[3]은 1990년대 초에 만들어진 강력한 통계 및 데이터 시각화 언어로써, 과학 계산 커뮤니티에서 가장 널리 사용되는 프로그래밍 언어 중 하나이다. 그러나 R의 강력한 기능에도 불구하고 2012년 Shiny 패키지[4]가 개발되기 이전에는 웹 애플리케이션을 구축하는 연구에서는 거의 사용되지 않았다. Shiny는 웹 애플리케이션을 개발하기 위해 기존의 웹 개발 언어인 HTML, CSS, 또는 JavaScript 등을 사용할 필요 없이 대규모 비즈니스 웹 애플리케이션 구축을 매우 쉽고 강력하게 만들어 줄 수 있다.

R/Shiny는 데이터 시각화나 데이터 검색을 위한 경량 웹 애플리케이션 구축뿐만 아니라 엄청난 수의 사용자 트랜잭션이 발생하는 대규모 빅데이터 서버를 구축하는 데에도 활용될 수 있다. R은 기본적인 plotting 기능, 격자(Lattice) 기능, ggplot2 라이브러리 등을 포함하여 다양한 시각화 기능이 제공되고, 특히 ggplot2 라이브러리는 과학계에서 가장 널리 사용되는 그래픽 라이브러리 중 하나로 복잡한 분석 데이터를 매우 쉽게 시각화할 수 있다. 하지만 고수준의 plotting을 위해서는 여전히 R의 문법과 기능들에 익숙해야 하고 이는 일반 사용자에게 여전히 큰 과제가 될 수 있다. 그럼에도 불구하고 R은 현재 Shiny의 혜택으로 더 많은 대화형 데이터 시각화 응용 프로그램에서 사용되고 있고 더 많은 사용자들이 R의 강력한 시각화 기능을 즐길 수 있게 되었다.

강력한 통계 및 시각화 도구 외에도 R의 또 다른 장점은 빅데이터 연구 거의 모든 분야의 데이터 분석을 위해 수만 개의 패키지를 무료로 사용할 수 있다는 것이다. 이러한 장점은 대화형 웹 서버 구축 시 R/Shiny 패키지를 적용함으로써 더욱 강력해질 수 있다. 한편, R/Shiny는 복잡한 데이터베이스를 구축하는 데에도 사용될 수 있다. MySQL은 데이터베이스 시스템 개발에 가장 널리 사용되는 오픈 소스 데이터베이스 관리 시스템으로써[7], R의 장점은 MySQL 사용을 위해 DB 인터페이스 역할을 지원하는 RMySQL 패키지의 도움으로 쉽게 DBMS를 활용할 수 있다. 간혹, 다수 사용자 액세스를 위한 데이터베이스 시스템 구축이 아닌 경우 MySQL 대신 R에서 지원되는 sqldf 패키지를 사용하여 SQL 언어를 직접 코딩할 수도 있다[3].

III. 금융 조기경보시스템 개발

3.1 EWS 구조

EWS 구조는 그림 2와 같이 데이터 층(Data layer), 비즈니스 층(Business layer), 그리고 프리젠테이션 층(Presentation layer)의 3계층 구조로 구현되어 있다. 데이터 층에서는 금융사와 자료제공 기관 등으로부터 데이터가 수집되어 업무보고서 DB와 금융시장통계 DB를 구축하고, 이 데이터를 가공하여 조기경보 DB와 메타 DB를 구축한다.

본 시스템은 R/Shiny와 MySQL과 같은 오픈 소스 플랫폼을 사용하고 워크플로우 자동화 및 시각화 웹 애플리케이션 개발을 연구 목표로 삼고 있다. 금융사나 정보제공자로부터 입수한 데이터는 Excel 파일로 저장되어 있다. 이 Excel 파일들은 워크플로우 엔진을 통해 자동적으로 Business Report DB와 Financial Statistics DB를 구축하고, 이 DB들을 이용하여 최종적으로 Early Warning DB를 구축한다. 이를 위해 Trigger와 Stored Procedure 기반 DB 기법[8]과 R 데이터 분석/처리 기능이 사용된다. 특히 본 연구에서는 조기경보 예측 작업을 위해 통계 처리에서 강력한 기능을 제공하고 있는 R 프로그램으로 작성하였고 이 프로그램은 EWS DB와 연계되어 작동된다.

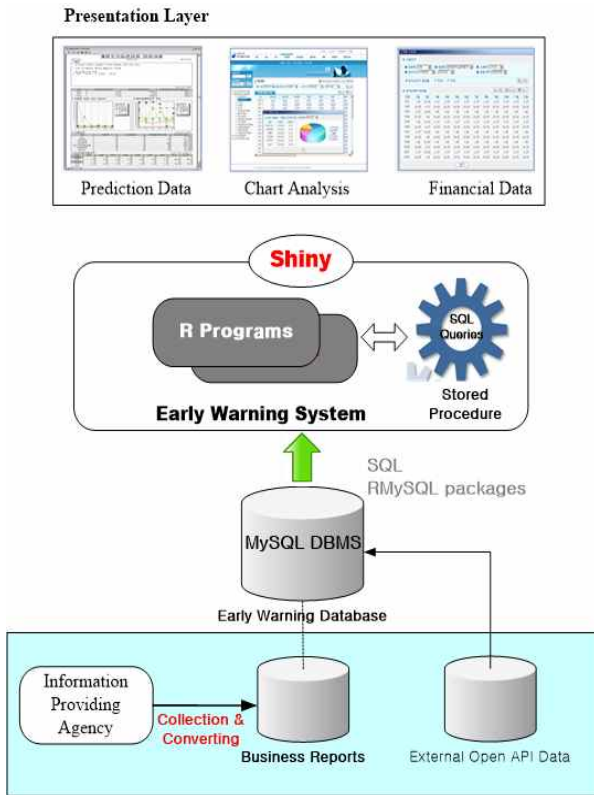


그림 2. EWS 전체 구조
Fig. 2. Overview architecture of EWS

새롭게 개발된 EWS는 프리젠테이션 층을 위하여 최신의 R/Shiny 기법을 사용하여 웹 기반으로 개발되어, 언제 어디서나 쉽게 시각화 데이터를 검색하고 활용할 수 있도록 개발한다. R/Shiny는 웹 애플리케이션 프레임워크로 HTML, CSS, 자바스크립트 등 웹 언어를 사용하지 않고 R 언어만으로 웹 애플리케이션을 만들 수 있다. Shiny로 만든 앱은 최신의 웹 프레임워크를 채용하고 있는 반응성 (Reactivity) 모드를 디폴트로 사용하기 때문에 사용자가 입력한 값의 변화에 따라 결과가 즉각적으로 반응하는 역동적(Dynamic)인 출력을 제공해 준다. 정리해 보면, EWS는 R 프로그램과 Shiny로 오픈 소스 기반의 새로운 시스템으로 개발되었으며, 최신의 분석기법(예, 딥러닝 기술 등)을 사용하여 정확성과 효율성을 향상시킨다. 또한 웹 기반 반응형 GUI(Graphic User Interface)로 개발되어 쉽고 편리하게 활용할 수 있도록 개발되었다. DBMS는 오픈 소스인 MySQL DBMS를 사용하였고 다른 타사 제품의 DBMS로도 변경 가능하다.

3.2 웹 애플리케이션 구현 기법

R의 인기를 뒷받침하는 주요 원동력은 Shiny 패키지를 포함하여 R 커뮤니티에서 제공하는 수많은 오픈 소스 패키지라 할 수 있다. 현재 수만 개의 R 패키지가 CRAN[9], GitHub[10], rOpenSci[11] 등에서 무료로 제공되고 있다. 이러한 패키지들은 R/Shiny 웹 애플리케이션 소스 코드에 직접 적용되어 그래픽 인터페이스를 개선하거나 서버와 클라이언트 간의 통신을 강화하는 데 사용될 수 있다. 본 프로젝트를 위하여 사용된 R 패키지들은 그림 3과 같다.

- shiny, dplyr, shinydashboard, htmlWidgets,
- shinyWidgets, ggplot2, tidyverse, DT, rlang,
- rsconnect, colourpicker, ggExtra, ggcorrplot, miniUI,
- reactR, reactable, reshape2, shinyhelper, forcats,
- scales, plotly, repmis, anytime, ROCR, data.table,
- jstable, ggpubr, recipes, ModelMetrics, dplyr,
- stringr, MASS, caret, ggh4x, parallel, xgboost,
- randomForest, reshape2, pROC, shinydashboardPlus,
- patchwork, Metrics, sqldf

그림 3. EWS 개발을 위해 사용된 R 패키지
Fig. 3. Used R packages for developing EWS

CRAN에서 가장 많이 사용되고 있는 dplyr 패키지는 R에서 데이터의 용이한 조작을 위해 만든 것이다. 우리는 dplyr에서 select, filter, mutate, summarize, group_by와 같은 함수들을 사용하였다. 대시보드는 특정 목표와 관련된 지표나 주요 데이터를 추적, 분석, 그리고 시각화하기 위해 대화형 접근 방식을 제공하는 GUI이다. 대시보드는 비즈니스, 산업, 그리고 과학 분야에서 광범위하게 활용되어 전략적 의사 결정이나 데이터 분석을 지원한다. 그동안 많은 대시보드들이 개발되었는데, 대표적으로 Google Data Studio[12], DashThis[13], IBM Watson Analytics[14], 그리고 R/Shiny 등이 있다. 이들 도구 중 R/Shiny는 대시보드를 만드는 최고의 오픈 소스 도구 중 하나로써, R/Shiny 대시보드는 shinydashboard, flexdashboard, 그리고 semantic.dashboard를 포함한 세 가지 R 패키지를 제공한다.

Shinydashboard와 flexdashboard는 각각 bootstrap 그리드 레이아웃과 CSS flexbox 레이아웃을 기반으로 구성되고, semantic.dashboard는 semantic UI를 기반으로 개발되었다. Flexdashboard는 R Markdown을 활용하여 R 이외의 코드를 최소화한다는 장점이 있고, shinydashboard와 semantic.dashboard는 Shiny용 템플릿을 제공하여 멋진 대시보드를 더 쉽게 만들 수 있도록 지원한다. Shinydashboard는 header, sidebar, body 3가지로 구성되어 있는데 각각 DashboardHeader(), DashboardSidebar(), DashboardBody() 함수를 사용하여 구축할 수 있다. 이들 3개 함수는 대시보드 페이지를 생성하는 DashboardPage() 함수 내에 배치되어야 한다. HtmlWidgets 패키지는 JavaScript 언어로 개발된 다양한 라이브러리를 Shiny에 임베딩함으로써 상호작용 기능을 추가할 수 있고, shinyWidgets을 사용함으로써 R/Shiny 애플리케이션의 GUI 기능을 향상시키며 R/Shiny에서 사용할 수 없는 다양한 위젯을 제공받을 수 있다. 예를 들면, 스위치 버튼을 사용하여 R/Shiny의 기존 확인란을 대체하고 슬라이더 텍스트를 사용하여 R/Shiny의 선택 입력을 대체할 수 있다.

ggplot2는 비즈니스 분야에서 가장 널리 알려진 시각화 관련 책 “Grammar of Graphics”의 디자인 철학을 기반으로 Hadley Wickham이 만든 패키지이다. ggplot2의 기본 구성요소는 크게 layer, scale, geom으로 나뉘며, layer는 층별로 그림을 그리겠다는 선언이고, scale은 양적 요소를 그리고, geom은 그림으로 표현된 물체를 의미한다. tidyverse는 데이터 분석에 유용한 여러 패키지들을 한데 모아 뒀는데, 데이터 시각화를 위한 ggplot2, 데이터 프레임 분석 및 처리를 위한 dplyr, 문자열 처리를 위한 stringr, 이외에도 readr, tibble, tidyr 등 많은 패키지들이 포함되어 있어 이를 활용하였다.

3.3 R/Shiny 애플리케이션 배포

R/Shin의 폭발적인 인기에도 불구하고 웹 기반 운영 환경으로 앱을 배포(Deployment)하는 것은 매우 번거롭고 복잡한 일여서 이는 종종 비IT 연구가들에게는 힘든 작업 중 하나이다. 더욱이 기본 Shiny 앱은 하나의 앱에 여러 사용자가 동시에 액

세스하는 것을 허용하지 않기 때문에 이러한 문제를 회피하는 한 가지 솔루션은 Shiny 애플리케이션을 호스팅하는 전용 웹 사이트인 Shinyapps.io[15]를 사용하는 것이다. 이는 사용자가 방화벽이나 Shiny 서버 플랫폼을 구성할 필요가 없기 때문에 Shinyapps.io에서 R/Shiny 애플리케이션을 배포하고 관리하는 것은 매우 쉬운 일이다. Shinyapps.io에서는 각 계정에 대한 토큰과 패스워드가 자동으로 생성되며, rsconnect 패키지를 사용하여 로컬 컴퓨터에서 Shinyapps.io로 R/Shiny 응용 프로그램을 업로드하고 배포할 수 있다. 이 사이트를 이용하면 최대 5개까지 R/Shiny 애플리케이션들을 무료로 배포할 수 있지만, 더 많은 애플리케이션을 배포하려면 유료 계정이 필요하다. Posit Connect[16]는 RStudio 클라우드에서 R/Shiny 애플리케이션을 호스팅하기 위해 제공되는 또 다른 서비스이다. Shinyapps.io와 비교하여 Posit Connect는 훨씬 더 많은 기능을 제공하지만 유료 사용자들에게만 공개되는 단점이 있다.

한편, 자체적으로 웹 서버를 구축하여 각 기관이 개발한 R/Shiny 응용 프로그램들을 배포하려 한다면 Shiny Server[17] 프로그램 설치가 필요하다. Shiny Server는 64비트 Linux 운영 체제에서만 실행되는 무료 오픈 소스로서, Shiny Server가 설치된 단일 Linux 서버의 방화벽 뒤에 여러 개의 R/Shiny 애플리케이션들을 배포할 수 있다. R/Shiny 애플리케이션 소스는 일반적으로 Shiny Server에서 생성된 /srv/shiny-server 디렉터리에 업로드되어야 하고, /etc/shiny-server/shiny-server.conf 파일을 편집하여 소스 위치나 포트 설정 등을 구성할 수 있다. R/Shiny 응용 프로그램은 Linux 서버가 기동되면 서버의 IP 주소와 포트 그리고 프로그램 디렉터리로 구성된 URL(Uniform Resource Location) 주소로 액세스 된다. Shiny Server 외에도 ShinyProxy[18]는 자체 Linux 서버에 R/Shiny 애플리케이션들을 배포하기 위한 또 다른 무료 오픈 소스 솔루션으로써 Shiny 앱과 Docker[19] 간의 통신에 사용되며 사용자 인증과 같은 추가 기능도 제공한다. Shiny Server는 각각의 R/Shiny 애플리케이션에 대해 단일 프로세스만 지원하는 반면에 ShinyProxy는 멀티 프로세스를 지원하므로 계산 집약적인 애플리케이션일 경우 ShinyProxy가 더 많은 이점이 있을 수 있다.

우리는 VMware Workstation[20]를 사용하여 윈도 환경에서 Linux 가상 머신(Virtual machine)을 실행하였다. 우분투(Ubuntu)[21]는 가장 인기 있는 리눅스 무료 배포판 중 하나로, 우분투에 Docker를 설치하면 환경에 구애받지 않고 애플리케이션들을 신속하게 배포 및 확장할 수 있다. Docker는 애플리케이션들을 신속하게 구축, 테스트, 그리고 배포할 수 있는 소프트웨어 플랫폼으로, Docker는 소프트웨어를 컨테이너라는 표준화된 유닛으로 패키징하고, 이 컨테이너에는 라이브러리, 시스템 도구, 코드, 런타임 등 소프트웨어를 실행하는 데 필요한 모든 것이 포함되어 있다. 한편으로, Shiny Server와 Docker 외에도 Nginx 설치가 필요한데, Nginx는 웹 앱을 쉽고 안전하게 관리해 주는 웹 서비스 프로그램이다.

웹 애플리케이션에서는 웹 브라우저가 최종 사용자에게 그래픽 인터페이스를 제공하는 프론트엔드 역할을 한다. 서버 측에서는 컨테이너화된 소프트웨어(예를 들면, Shiny App, Shiny Server, Nginx)가 Docker 환경 내에 배포된다. 여기서 Nginx는 사용자 요청을 Shiny Server로 전달하는 역방향 프록시 역할을 하며 Shiny Server는 필요한 Shiny 애플리케이션들을 실행시킨다. Shiny Server는 여러 사용자가 동일한 Shiny 앱에 액세스하려고 할 때 수요에 맞게 컨테이너화된 Shiny 앱을 자동으로 복제하고 작업이 완료되면 삭제하는 방식으로 멀티 사용자들을 관리한다. 따라서 앞서 언급한 것처럼 기본 Shiny 앱의 단일 사용자 액세스 문제를 해결할 수 있다.

3.4 EWS 실행 화면

EWS 메뉴는 그림 4와 같이 데이터 처리, 탐색적 데이터 분석, 통계 모델링, 가시화 등으로 구성되어 있으며 이는 향후 수요자 요청에 따라 계속해서 하나씩 서비스가 추가될 예정이다. EWS 데이터베이스의 테이블 구성은 약 50개 정도의 속성으로 구성되어 있는데 이들 중 기본 데이터는 약 20개 정도로 구성되어 있다. 예를 들면, 날짜(date), 은행코드, CAR(Capital Adequacy Ratio), PPOP(Pre-Provision Operating Profit), House_Gap, Loan 등이 있다. 변수(Variable) 데이터는 약 30개 정도로 구성되어 있

며 기본 데이터를 사용하여 조기경보 모델에 사용할 수 있는 변수 모델을 계산한 값이라고 볼 수 있다.

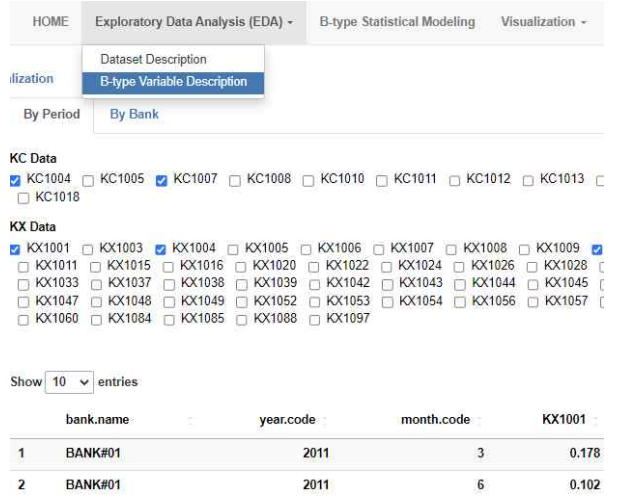


그림 4. EWS 전체 메뉴
Fig. 4. All menu for EWS

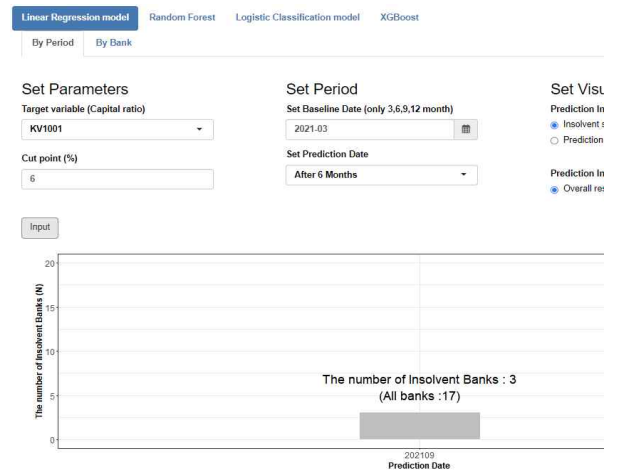


그림 5. 로지스틱 회귀분석을 수행한 결과
Fig. 5. Result of performing logistic regression

그림 5는 부실화된(Insolvent) 은행에 대한 로지스틱(Logistic) 회귀분석을 수행한 결과를 웹 브라우저를 통해 보여 주고 있다. 기간(Period), Cut point, After N months, 은행 타입 등을 마우스로 클릭하여 선택한 후 실행하면 그림과 같은 결과를 디스플레이 해 준다. 아래 그래프는 예측 결과로써 실제 부실화된 결과를 보여 주고 있다. 또한 그림 6의 탐색적 데이터 분석에서는 EWS의 중요한 속성 데이터라 할 수 있는 총자본(Equity), 총자산(Asset), 총부채(Liability), 대손비용(DBA, Bad Debt Allowance), 총

당금적립전이익(PPOP), House Price Gap에 대한 그림을 보여 주고 있다. 조만간 로지스틱 회귀분석을 기반으로 한 머신러닝 방법들, 예를 들면, XGBoost, Random Forest, Support Vector Machine, 그리고 Deep Neural Network Model 등 다양한 모델들이 개발될 예정이다.

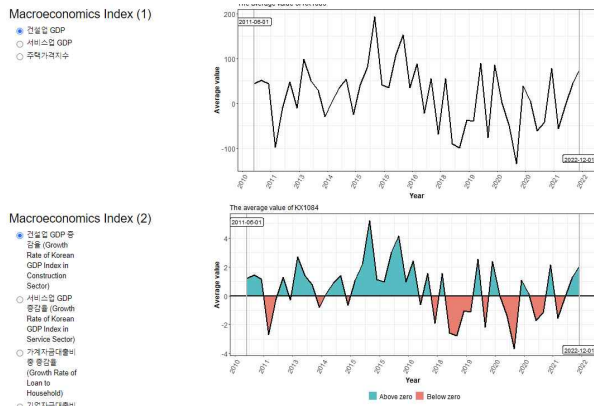


그림 6. 탐색적 데이터 분석 결과
Fig. 6. Result of exploratory data analysis

IV. 성능 분석

본 장의 목적은 기존의 전통적인 Java 기반 웹 애플리케이션[22]과 비교하여 본 논문에서 제안하는 R/Shiny 대화형 웹 애플리케이션의 우수성을 보이는 것이다. 각각의 시스템에 대한 효율성(eficiency)과 이에 따른 이용자 만족도(satisfaction)를 파악하기 위해 IT 전문가 3명과 금융 분야 전문가 3명을 대상으로 면담을 실시하였다. 이는 금융 EWS 분야에 대하여 일반 이용자들의 인식이 보편적이지 않기 때문에 이들을 대상으로 평가하는 양적 평가 방법은 한계가 있으므로, 국내 선행 전문가들을 직접 면담하고 그들이 진술한 면담 내용을 분석하는 질적 평가 방법을 채택하였다. 본 성능 분석의 기준인 효율성과 만족도는 다음과 같이 정의될 수 있다.

효율성(Efficiency): 이용자가 업무 경험을 바탕으로 본 시스템을 사용했을 경우 실제 활용 측면에서 효과가 있을 것으로 기대하는 정도

만족도(Satisfaction): 이용자가 시스템의 활용을 통해 느끼는 만족도

그림 7은 효율성에 대한 평가 항목별 평균 점수를 보여 주고 있다.

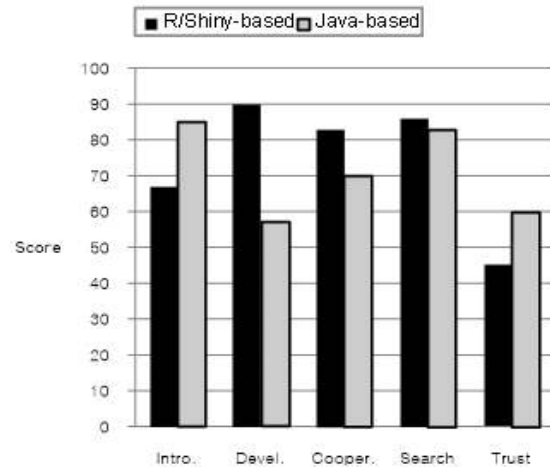


그림 7. 효율성 평가 점수
Fig. 7. Efficiency evaluation score

각 항목별로 자세히 살펴보면, ① 향후 연구 또는 업무 적용시 시스템 도입 여부는 Java 시스템이 높게 나왔는데, 이는 기존의 시스템이 대부분 Java 기반으로 개발되어 있으므로 새로운 솔루션 도입에 대한 부담감과 호환성 문제 등으로 기인한다. ② 시스템 개발 환경은 R/Shiny 시스템 점수가 매우 높는데, R/Shiny는 순수 R 코드로 복잡한 웹 애플리케이션을 쉽게 개발할 수 있으며, HTML, CSS, JavaScript 등 웹 기술에 익숙하지 않은 R 사용자에게 가장 큰 이점이 될 수 있기 때문이다. 또한 MySQL과 Apache 등을 사용하지 않고도 멀티 사용자를 위한 웹 서버를 구축할 수 있는 장점이 있기 때문이다. ③ 업무 협업 부분에서 R/Shiny는 다양한 분야의 전문가들이 자신만의 애플리케이션을 쉽게 개발할 수 있도록 함으로써 팀간 협업과 재사용 가능한 연구를 촉진할 수 있을 것으로 평가되었다. R/Shiny를 사용한 웹 애플리케이션 개발의 또 다른 장점은 빅데이터 연구의 거의 모든 분야를 포괄하는 수만 개의 R 패키지를 무료로 사용할 수 있다는 점이며, 이는 빅데이터 활용 및 분석에 효율적으로 활용될 수 있을 것이다. ④ 정보 검색시 Java 시스템은 시스템의 작동 방식을 이해해야 하고 원하는 검색 결과를 얻기 위해 복잡한 사용 방법을 배워야 하는 어려움이 있다.

반면에 R/Shiny 시스템은 직관적 방식으로 검색은 쉬우나 반응성 모드를 디폴트로 사용하기 때문에 사용자가 입력값 변경 시 바로 결과에 반영되는

방식에 아직 익숙하지 않아 조금 불편함을 느끼는 것으로 나타났다. ⑤ 고도의 전문영역을 다루는 통계 분야 EWS 적용에 대한 신뢰성 항목에서는 R/Shiny 시스템에 대한 회의적인 의견을 보였다. 이는 이용자의 편의성도 중요하지만, 금융 분야는 무엇보다 데이터의 신뢰성이 중요한 시스템으로 최신 신기술에 대한 신뢰성을 검증할 방안이 심도 있게 제시될 필요가 있다. 전반적으로 R/Shiny 시스템이 Java 시스템보다는 도입과 신뢰성 문제만 제외하곤 좋은 평가를 얻은 것을 알 수 있다.

그림 8은 만족도를 각 평가 항목별로 보여주고 있다. 본 연구에서는 만족도에 영향을 주는 요인으로 필요한 서비스 제공 여부, 콘텐츠 접근성, 액세스 속도, 그리고 소프트웨어 업그레이드를 선택하였다. 각 항목별로 자세히 살펴보면, ① 필요한 서비스 제공 여부는 R/Shiny 시스템이 Java 시스템보다 높았는데 고수준의 대시보드 구현 방식으로 다양한 그래픽 기반 정보를 제공하고, 수많은 무료 패키지 제공으로 활용도가 높게 평가되었다. 그러나 데이터 세트를 처리하기 위해 R은 전체 데이터 세트를 한번에 메모리에 읽고, 생성된 개체가 메모리에 전체 저장된다. 따라서 R/Shiny는 서버 측에서 매우 큰 데이터 세트를 처리하거나, 사용자 측에서 매우 큰 파일을 업로드해야 하는 애플리케이션 개발에는 적합하지 않을 수 있다. ② 콘텐츠 접근성에 대해 R/Shiny 시스템은 Java 시스템보다 훨씬 높게 나타났다. R 프로그램의 강력한 시각화 기능은 R/Shiny의 장점을 더욱 강화하고, CSS, HTML, 그리고 JavaScript 코드를 R/Shiny 애플리케이션 코드에 임베딩하여 R/Shiny 애플리케이션의 모양과 기능을 추가로 사용자 정의할 수 있는 기능 때문이다. ③ 액세스 속도는 Java 시스템의 만족도가 높게 나왔는데, 현재 R/Shiny 시스템에서는 분산 컴퓨팅이나 동시성 처리와 같은 기능은 아직 잘 지원되지 않는 것에 기인한다. ④ 소프트웨어 업그레이드는 신기술 수용, 새로운 버전관리, 교체 비용 등으로 평가되는데 업그레이드에 관해서는 Java 시스템이 높은 점수로 평가되었다. R/Shiny 라이브러리는 현재 계속 버전 업그레이드되고 있는 시스템으로써, R 패키지에 구현된 R 함수의 매개변수가 업데이트 중에 변경될 수 있으므로 종속 R 패키지가 업데이트될 때

R/Shiny 애플리케이션이 경고 없이 충돌될 수 있다. 또한 우수한 자동화 테스트 도구가 부족하여 R/Shiny 애플리케이션을 디버깅하는 것도 어렵다.

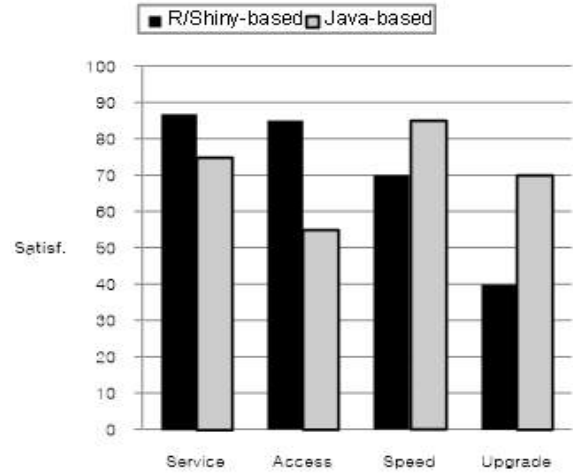


그림 8. 만족도 평가 점수
Fig. 8. Satisfaction evaluation score

EWS에서 가장 중요한 것은 부실한 은행을 정확히 판별할 수 있는 예측력 높은 정교한 시스템을 개발하는 것이다. 개발된 시스템의 예측 품질을 측정하기 위해 한국 글로벌 금융위기 기간인 2008년도 경제지표에 대해 총 2,206개 행을 추출하여 실험을 실시하였다. 그림 9는 실험 데이터에 대한 모든 값들의 표현이 쉽지 않아서 구간 이동하면서 측정된 실제값과 예측값의 일부값을 보여주고 있다.

가로축은 실험 순서를, 세로축은 R_{t+1} (단위: %)를 나타낸다(2장에선 4% 이하면 부실로 판단). 이를 살펴보면 우리의 예측값이 실제값과 상당히 일치하고 있음을 보여주고 있다.

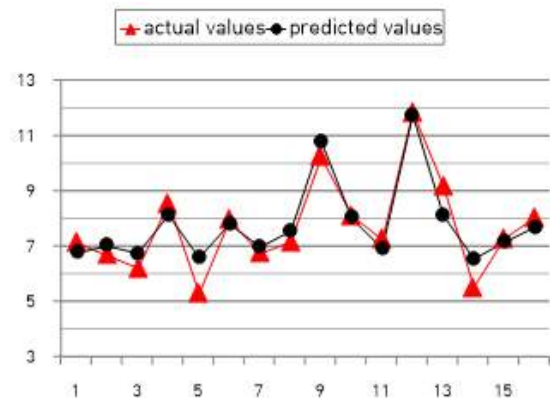


그림 9. 실측값 대 예측값
Fig. 9. Actual vs. predicted values

V. 결 론

본 논문에서는 현재 일부 수작업으로 수행되고 있는 EWS 업무 전체를 자동화하여 시스템의 안전화 및 기능화를 구현하였고, R/Shiny와 MySQL 등 오픈 소스 플랫폼을 사용하여 개방형 워크플로우 및 대쉬보드 가시화를 구현하였다. 개발된 EWS는 최근 통계 분야에서 널리 사용되고 있는 R 프로그램을 사용하여 개발하였고 이 프로그램은 MySQL DB와 연계되어 운용된다. 또한 프리젠테이션 층 GUI 개발을 위하여 Shiny 패키지를 활용하여 대화형 웹 기반으로 구현하였다. 그러나 R/Shiny 앱은 동시 멀티 유저 환경을 허용하지 않기 때문에 Shiny Server 프로그램을 설치하여 자체 전용 웹 사이트를 구축하였고, 이는 기본 Shiny 앱의 멀티 사용자 액세스 문제를 해결할 수 있었다. 본 연구에서는 관련 전문가 6명을 대상으로 면담 내용을 분석하는 질적 평가 방식으로 기존의 전통적인 Java 기반 애플리케이션과 본 논문에서 제안하는 R/Shiny 기반 애플리케이션의 성능을 분석하였다. 분석 결과 기존 시스템보다 제안 시스템이 효율성과 만족도 측면에서 보다 더 우수함을 보였다. 또한 시스템 예측 품질을 측정하기 위해 경제지표 데이터에 대한 정량적 실험을 통해 예측값의 정확성을 보여 주었다.

R/Shiny는 R 사용자가 대화형 웹 애플리케이션을 구축할 때 채택할 수 있는 사용하기 쉽고 강력한 프레임워크로써, 직관적이고 반응적인 빅데이터 분석 및 데이터 시각화 개발을 촉진시킬 것으로 기대된다. 그러나 기존 시스템들이 대부분 Java를 기반으로 개발된 상황에서 R/Shiny와 같은 새로운 솔루션 도입에 대한 부담감과 호환성 문제 등은 앞으로 해결되어야 할 주요한 과제이다.

References

- [1] International Monetary Fund, "Global financial stability report: Financial and climate policies for a high-interest-rate era", IMF Publications Services, pp. 1-51, Oct. 2023.
- [2] E. Allaj and S. Sanfelici, "Early Warning System for Identifying Financial Instability", *International Journal of Forecasting*, Vol. 39, No. 4, pp. 1777-1803, Dec. 2023. <http://dx.doi.org/10.2139/ssrn.3738936>.
- [3] K. Medeiros, "R programming fundamentals", Packt Publishing, Sep. 2018.
- [4] C. Beeley, "Web application development with R using Shiny", Packt Publishing, Jan. 2016.
- [5] J. H. Lang, T. A. Peltonen, and P. Sarlin, "A framework for early-warning modeling with an application to banks", *European Central Bank, ECB Working Paper Series No. 2182*, Oct. 2018. <https://doi.org/10.2866/25674>.
- [6] S. D. Jeung, "Toward a More Effective Early Warning System: Focus on FSS Early Warning System for Korea's Financial Services Industry", *Journal of Korean Economic Development*, Vol. 14, No. 2, pp. 113-135, Dec. 2008.
- [7] S. Tongkaw and A. Tongkaw, "A comparison of database performance of MariaDB and MySQL with OLTP workload", *2016 IEEE Conference on Open Systems (ICOS)*, Langkawi, Malaysia, pp. 525-527, Oct. 2016. <https://doi.org/10.1109/ICOS.2016.7881999>.
- [8] W. Da-sheng and W. Sheng-yu, "Dynamically maintain the teaching examples of triggers and stored procedures about the course of database application", *2010 2nd International Conference on Education Technology and Computer*, Shanghai, China, pp. 525-527, Jun. 2010. <https://doi.org/10.1109/ICETC.2010.5529193>.
- [9] Institute for Statistics and Mathematics, "The comprehensive R archive network", <https://cran.r-project.org> [accessed: Apr. 12, 2024]
- [10] A. Decan, T. Mens, M. Claes, and P. Grosjean, "When github meets CRAN: An analysis of inter-repository package dependency problems", *Proc. 23rd Inter. Conf. on Software Analysis, Evolution, and Reengineering*, Osaka, Japan, Mar. 2016. <https://doi.org/10.1109/SANER.2016.12>.
- [11] C. D. Boettiger, S. Chamberlain, E. Hart, and K.

Ram, "Building Software, Building Community: Lessons from the rOpenSci Project", Journal of Open Research Software, Vol. 3, No. 1, Nov. 2015. <https://doi.org/10.5334/jors.bu>.

- [12] Google, "Google data studio: Put the power of your marketing insights in everyone's hands", https://services.google.com/fh/files/misc/data_studio_product_overview.pdf [accessed: Apr. 12, 2024]
- [13] Dashthis, <https://dashthis.com/> [accessed: Apr. 12, 2024]
- [14] Element61, "IBM Watson Analytics", <https://element61.be/en/competence/ibm-watson-analytics> [accessed: Apr. 12, 2024]
- [15] Posit, "Share your Shiny applications online", <https://www.shinyapps.io/> [accessed: Apr. 14, 2024]
- [16] Posit, <https://posit.co/products/enterprise/connect/> [accessed: Apr. 14, 2024]
- [17] Posit, <https://posit.co/products/open-source/shinyserver/> [accessed: Apr. 14, 2024]
- [18] Open Analytics, <https://www.shinyproxy.io/> [accessed: Apr. 14, 2024]
- [19] Docker, <https://namu.wiki/w/Docker/> [accessed: Apr. 14, 2024]
- [20] Broadcom, <https://www.vmware.com/content/vmware/vmware-published-sites/us/products/workstation-player/> [accessed: Apr. 14, 2024]
- [21] Canonical Ubuntu, <https://ubuntu.com/desktop> [accessed: Apr. 14, 2024]
- [22] Bank of Korea, "Reporting work handbook for foreign exchange information system", BOK Working Paper Series, Bank of Korea, Dec. 2021.

저자소개

이 용 주 (Yongju Lee)



1985년 : 한국과학기술원
정보검색전공(공학석사)
1997년 : 한국과학기술원
컴퓨터공학전공(공학박사)
1998년 8월 ~ 현재 : 경북대학교
IT대학 컴퓨터학부 교수
관심분야 : 링크드 데이터, 시맨틱
웹, 빅데이터, 지식 그래프

김 영 민 (Young Min Kim)



2005년 : 연세대학교
응용통계학(통계학 석사)
2012년 : 아이오와주립대학교
통계학(통계학 박사)
2016년 3월 ~ 현재 : 경북대학교
자연과학대학 통계학과 부교수
관심분야 : 데이터 정보보호,
방사선 의학, 부츠트랩 방법