Credit Card Default Prediction
By: Tyler Bain
Advisor: D.C. Grant | School of Engineering and Technology

Abstract
Credit card defaulting is the main problem that credit card companies face. If a person can't pay their credit card bill, then the credit card company is on the line to pay that money. It is important for the company's bottom line that they don't give money to those who might be unable to pay it back. My program seeks to solve this by being able to predict whether a person can pay back their debts.

Introduction
There are several Machine Learning models that could assist in default prediction, but Support Vector Machines fit this situation perfectly. They work well in any amount of dimensions which is great for the dataset I used with its 25 features. In a couple minutes, a Support Vector Machine can correctly classify whether a user will be able to pay their bill or default.

Methods
For this project, I used Microsoft Excel to clean some of the data and format it. From then I used IntelliJ and RStudio to write my program and tune the support vector machine. With JRI API, I was able to use R within the Java application.

Results
Using a JAVA and R, the support vector machines had an accuracy rate of about 86%. Out of the predictions that were incorrect, a majority were false positives which are superior to false negatives because a credit card company would rather not loan money than loan and lose it. All in all, the results were impressive and could be improved further with a tune of the svm.

Conclusions
This project pushed my programming and machine learning skills further than I ever expected. I am satisfied with the results of the prediction but will continue to work on the project to improve upon it. I can confidently say I am a better programmer because of this project and am better suited for a career in the Data Analytics and Data Science field.

Future Directions
I will continue to work on this project after the quarter ends to make the predictions more accurate and quicker. I would like to try running the predictions on a virtual environment to improve the performance of the support vector machines. At this point I am limited by the power of my personal computer, but this would be one solution. Also, I will test different machine learning models to see what performs the best in the lowest amount of time. Random Forests are one option, but I still need to try a few out.

Acknowledgements
I would like to thank Professor Grant for the opportunity and the guidance to complete this project. I would also like to thank James Martin and Randy Strobel for helping me on this project and many others whenever I got stuck.