C964 - Computer Science Capstone

Jenna Brannan

ID #011245264

WGU Email: jbra933@wgu.edu

Date: 12/21/2024

Part A: Letter of Transmittal

Date: December 19, 2024

Chris Reynolds

Chief Operating Officer

Iron Financial

789 Copper Road

Provo, UT, 84601

Subject: Proposal for Credit Card Approval Prediction System Implementation

Dear Mr. Reynolds,

I am excited to propose a project that will improve Iron Financial's credit card application process. Over the past two years, the company has experienced a 40% increase in credit card applications. While this growth demonstrates Iron Financial's success, it has also overwhelmed the current manual approval process. This method is slow, prone to human error, and difficult to scale, resulting in delays, customer dissatisfaction, and higher operational costs.

I would like to propose implementing a Credit Card Approval Prediction System powered by machine learning. This system will automate application reviews, reduce processing times, and ensure decisions are consistent, fair, and accurate. The predictive model will achieve at least 85% accuracy and integrate seamlessly with Iron Financial's existing systems. An interactive interface will provide real-time predictions, streamlining operations while improving customer experiences.

The system will use historical credit card application data from Iron Financial’s database. This data will include applicant demographics, credit history, financial metrics, and prior application outcomes. All data will be anonymized to ensure privacy and compliance with ethical and legal guidelines, such as GDPR and CCPA. Sensitive information, such as birthdates or social security numbers, will not be used. Any incomplete or inconsistent records will be addressed during data preprocessing to maintain the integrity of the dataset.

The project will follow the SCRUM methodology, which is a an iterative process that promotes collaboration. This methodology will ensure flexibility and responsiveness to stakeholder feedback throughout the development process. The system will be developed in five phases over eight weeks.

The project will require $23,000 to cover development, testing, and deployment costs. The system will have a positive impact on all stakeholders. Customers will experience faster and fairer application processing, while staff will benefit from reduced workloads and streamlined workflows. Decision-makers will gain actionable insights, allowing them to make data-driven decisions confidently. It will also give Iron Financial the tools to scale its credit card offerings while staying ahead of competitors.

This project has three key objectives:

* Automate the credit approval process to improve efficiency and reduce processing times by 50%.
* Deliver a predictive model that achieves at least 85% accuracy on test data.
* Reduce customer complaints about delayed processing by 30%.

The following hypotheses will guide the project:

* Automating the credit approval process will reduce the average processing time from 48 to less than 24 hours.
* As validated on test datasets, the predictive model will correctly classify at least 85% of applications.
* Automation will reduce customer complaints about processing delays by at least 30% within the first year of deployment.

As a data scientist, I bring years of experience working with machine learning systems. I am confident that this solution will benefit Iron Financial and its customers significantly.

Thank you for taking the time to review my proposal. I look forward to discussing this project further and welcome any questions.

Sincerely,



Jenna Brannan

Data Science Manager

Part B: Project Proposal Plan

Project Summary

Iron Financial is struggling to approve credit card applications promptly due to its reliance on a manual approval process. Over the past two years, the company has experienced a 40% increase in credit card applications, which has further strained this outdated system. The manual process is slow, inconsistent, and prone to errors, making it challenging to handle the growing volume of applications efficiently. These issues lead to decision delays, customer frustration, and higher operational cost**s.** The proposed Credit Card Approval Prediction System will solve these problems by automating the approval process using machine learning. This system will make decisions faster, more accurate, and fairer. Reducing manual work and improving efficiency will help modernize operations and improve customer satisfaction.

The primary users of this system will be the credit evaluation team and company decision-makers, who currently deal with delays and inefficiencies caused by the manual process. The new system will simplify their work by automating repetitive tasks, speeding up decision-making, and offering clear insights through a simple interface. Customers will also benefit because their applications will be processed faster and more fairly.

The project will implement a machine learning model using a Random Forest Classifier and an interactive Jupyter Notebook for real-time predictions. By automating the current manual process, the system will enhance efficiency, minimize delays, and increase transparency. This will enable Iron Financial to deliver improved customer service while optimizing resource utilization.

Data Summary

The data needed for this project will come from historical credit card application records provided by Iron Financial. The data will go through several steps to prepare it for the machine learning model, including cleaning, normalizing, and creating new features. Significant predictors like income, debt, and prior defaults will be identified during the analysis to improve the system's accuracy. This data fits the project well because it provides a detailed look at the factors influencing credit decisions. To protect privacy and meet legal requirements like GDPR and CCPA, all personal information will be anonymized, and sensitive details like birthdates and social security numbers will not be included. Any missing or incomplete records will be fixed or removed to ensure reliable and consistent data.

Examples of the types of data that will be used include:

* Gender (str)
* Age(int)
* Income (float)
* Debt(float)
* Credit Score (int)
* Employment Status (str)
* Length of Employment (int)
* Marital Status (str)
* Ethnicity (str)
* Prior Defaults (str)
* Driver’s License Status (str)
* Citizenship Status (str)
* Zip Code (int)
* Approval or Denial Outcome (str)

Implementation

This project will follow the SCRUM methodology, an Agile framework that focuses on flexibility, collaboration, and iterative progress. SCRUM is well-suited for this project because it allows for continuous feedback from stakeholders and ensures that each project phase effectively builds on the previous one. Deliverables include:

* A trained and validated Random Forest Classifier capable of predicting application outcomes with at least 85% accuracy.
* A user-friendly, interactive Jupyter Notebook dashboard for real-time predictions and insights.

The implementation will be carried out over five sprints across an eight-week timeline. Each sprint will focus on a specific part of the project:

Sprint 1: Planning and Requirements Gathering (1 week)

* Define clear project goals and success metrics.
* Identify and prepare the data sources to be used.

Sprint 2: Data Preprocessing and Analysis (2 weeks)

* Clean and preprocess the raw data to ensure it is consistent and ready for use.
* Perform exploratory data analysis to identify key predictors, such as income, debt, and prior defaults.

Sprint 3: Model Development (2 weeks)

* Train and fine-tune a Random Forest Classifier for predicting application approvals or denials.

Sprint 4: Interface Development (2 weeks)

* Build an interactive dashboard in Jupyter Notebook to allow users to make real-time predictions.
* Include visualizations and widgets to make the interface easy to use.

Sprint 5: Deployment and Feedback (1 week)

* Deploy the system in a secure local environment for end-user testing.
* Gather feedback from stakeholders and make necessary improvements.

Timeline

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Milestone or deliverable | Duration  (hours or days) | Projected start date | Anticipated end date | Resources |
| Sprint 1: Planning | 1 week | December 21, 2024 | December 27, 2024 | - Data Scientist: 20 hours  - Machine Learning Engineer: 10 hours  - Data Analyst: 10 hours |
| Sprint 2: Data Preprocessing | 2 weeks | December 28, 2024 | January 10, 2025 | - Data Scientist: 20 hours  - Machine Learning Engineer: 20 hours  - Data Analyst: 40 hours |
| Sprint 3: Model Development | 2 weeks | January 11, 2025 | January 24, 2025 | - Data Scientist: 30 hours  - Machine Learning Engineer: 50 hours |
| Sprint 4: Interface Development | 2 weeks | January 25, 2025 | February 7, 2025 | - Data Scientist: 30 hours  - Machine Learning Engineer: 20 hours  - Quality Assurance Analyst: 20 hours |
| Sprint 5: Deployment and Feedback | 1 week | February 8, 2025 | February 14, 2025 | - Data Scientist: 10 hours  - Machine Learning Engineer: 10 hours  - Quality Assurance Analyst: 20 hours |

Evaluation Plan

The Credit Card Approval Prediction System will undergo rigorous validation and verification at every stage of development to ensure it meets both project requirements and customer needs:

Data Validation:

* Raw data will be thoroughly cleaned and checked for inconsistencies, missing values, and outliers.
* Statistical analyses and visualizations will verify data quality and identify anomalies.

Model Testing:

* The machine learning model will be evaluated using cross-validation to measure its accuracy, precision, recall, and F1 score.
* Performance metrics will be compared against the 85% accuracy objective, ensuring the model reliably predicts outcomes.

Interface Usability Testing:

* Stakeholders will test the Jupyter Notebook dashboard to verify ease of use, responsiveness, and clarity. Feedback from users will guide refinements to ensure the interface aligns with their expectations and operational needs.

Stakeholder Feedback and Comparison:

* Predictions made by the system will be compared with historical outcomes and manual decisions to ensure consistency and fairness.
* Regular stakeholder reviews during sprints will validate the system’s alignment with organizational goals.

Post-Deployment Monitoring:

The system’s effectiveness will be evaluated through continuous monitoring of the following key metrics:

* Average Processing Time:
  + Baseline: 48 hours under the manual process.
  + Objective: Reduce to less than 24 hours post-deployment.
  + Monitoring: Weekly reports comparing current average processing times to the baseline.
* Customer Complaint Rate:
  + Baseline: 100 complaints per month regarding delays.
  + Objective: Reduce complaints related to application delays by 30% within the first six months.
  + Monitoring: Monthly analysis of customer service records to identify trends in complaints.
* Model Accuracy:
  + Baseline: Achieved 92.13% accuracy on test data during development.
  + Objective: Maintain an accuracy rate of ≥ 85% during production, assessed through periodic retraining and validation on newly collected data.
* Approval Decision Consistency:
  + Evaluate consistency in decisions made for similar applicant profiles.
* Stakeholder Feedback Loop:
  + Regular meetings with stakeholders will collect qualitative feedback on system usability, reliability, and effectiveness. This feedback will guide updates and enhancements to the system.
* Scheduled Updates and Optimization:
  + Quarterly Model Evaluation: Retrain and validate the model using the latest application data to adapt to evolving patterns in credit applications.
  + Annual System Review: Conduct a comprehensive system performance review, addressing scalability and incorporating advanced machine learning techniques if necessary.

Resources and Costs

The project will leverage existing resources effectively, with minimal additional costs. The estimated expenses total $23,000. Here is the breakdown of needed resources and costs:

• Hardware Costs: $3,000

High-performance computing resources will support data preprocessing, model training, and testing locally. These resources are necessary to handle potentially large datasets and run computationally intensive machine learning models, such as the Random Forest Classifier.

• Labor Costs: $12,000

This accounts for the time and effort of the project over the estimated six-week development period.

Here is a breakdown of the estimated labor costs:

* Data Scientist (Lead): $50/hour for 110 hours, totaling $5,500.
  + Responsible for overseeing the project, managing model development, and implementing the Jupyter Notebook widget.
* Machine Learning Engineer: $45/hour for 100 hours, totaling $4,500.
  + Focuses on building and optimizing the predictive model, testing, and assisting with the widget integration.
* Data Analyst: $40/hour for 60 hours, totaling $2,400.
  + Handles data cleaning, exploratory analysis, and feature engineering.
* Quality Assurance Analyst: $35/hour for 40 hours, totaling $1,400.
  + Conducts testing to ensure the system works as intended and is easy to use.

• Software Costs: $0

The project uses open-source tools, including Python, Scikit-learn, Matplotlib, and Jupyter Notebook. These tools eliminate the need for additional licensing fees or proprietary software.

• Environment Costs: $0

The system will be deployed locally, avoiding ongoing costs for cloud hosting, external storage, or managed services. Maintenance and operational costs are expected to remain low.

• Miscellaneous and Contingency Costs: $8,000

This portion of the budget accounts for unexpected expenses, such as additional hardware, extended labor hours, or adjustments based on stakeholder feedback.

Part D: Post-implementation Report

Project Purpose

Iron Financial Services faced significant challenges in processing credit card applications due to a manual evaluation method. This method was time-consuming, error-prone, and unable to scale effectively with the 40% increase in applications over the last two years. Customers experienced delays and inconsistencies, leading to dissatisfaction, while employees struggled with increased workloads.

To address these issues, a Credit Card Approval Prediction System was developed. The system automated the approval process using a Random Forest Classifier, significantly reducing processing times and improving decision accuracy. It includes a Jupyter Notebook-based interactive interface that allows real-time predictions, enabling staff to quickly evaluate applications and make consistent, data-driven decisions.

Hypotheses Evaluated:

1. Automating credit approval will reduce processing times from 48 hours to under 24 hours.
   * Result: Accepted; average processing time reduced to 20 hours.
2. The predictive model will achieve at least 85% accuracy.
   * Result: Accepted; model achieved 92.13% accuracy on test data.
3. Automation will reduce customer complaints about delays by 30%.
   * Result: Accepted; customer complaints have been reduced to an average of 63 a month from 100 a month.

Data Summary

The raw data used for the Credit Card Approval Prediction System originated from historical credit card application records provided by Iron Financial Services.

The key features included:

* Demographic Information: Gender, Age, Ethnicity, Citizenship Status, Zip Code.
* Financial Metrics: Income, Debt, Credit Score.
* Application History: Prior Defaults, Employment Status, Length of Employment.
* Outcome: Approval Status (Approved or Denied).

The data was collected from Iron Financial’s internal application processing system, which logs applicant details and outcomes for credit card applications. To ensure comprehensive analysis, simulated data was not used; instead, the focus was on preprocessing of the existing dataset.

Data Processing and Management Throughout the Development Life Cycle

Sprint 1: Planning and Requirements Gathering

* Data Audit:
* Conducted an initial review of the dataset to assess completeness, identify missing values, and determine feature relevance.
* Key features such as Income, Debt, Credit Score, and Prior Defaults were identified as critical predictors.
* Data Preparation Plan:
* Outlined specific preprocessing steps, including handling missing values, encoding categorical features, and scaling numeric variables.
* Defined goals for data quality, including completeness and normalization benchmarks.

Sprint 2: Data Preprocessing and Analysis

* Removed duplicate records and irrelevant features, such as Zip Code, which had limited predictive value.
* Removed rows that included missing data
* Binary features (e.g., Gender, Married) were mapped to numeric values (e.g., Male = 1, Female = 0).
* Multi-category features like Ethnicity and EducationLevel were one-hot encoded for compatibility with the model.
* Feature Engineering: Created a derived metric, Debt-to-Income Ratio, to capture the relationship between debt and income, improving predictive power.
* Divided the cleaned dataset into training (70%) and testing (30%) subsets to train and validate the machine learning model.

Sprint 3: Model Development

* Model Training:
* Trained a Random Forest Classifier on the processed data to predict approval or denial outcomes.
* Split the dataset into training (70%) and testing (30%) subsets to evaluate performance.
* Hyperparameter Tuning:
* Used GridSearchCV to optimize model parameters
* Improved model’s ROC AUC score to .97 from initial results of .96

Sprint 4: Interface Development

* Interactive Dashboard:
* Developed a Jupyter Notebook widget-based interface for real-time predictions.
* Integrated widgets to input applicant details and provided instant feedback on application outcomes (Approved or Denied).
* Visualizations:
* Created charts to display feature importance, approval rates by different features, ROC graph, and a confusion matrix for model evaluation.

Sprint 5: Deployment and Feedback

* Local Deployment:
* Deployed the system in a secure local environment using Jupyter Notebook for immediate access by the credit evaluation team.
* Ensured compatibility with existing workflows to minimize disruption.
* Stakeholder Feedback:
* Collected input from end-users regarding system usability and performance.
* Refined the interface based on feedback to improve user experience and operational efficiency.
* Continuous Data Monitoring:
* Incorporated automated checks to identify missing values.
* Retrained the model with updated datasets to adapt to evolving applicant patterns quarterly.
* Data Storage and Security:
* Stored datasets securely in a local database, ensuring restricted access to authorized personnel only.
* Annually reviewed updated regulations and best practices regarding storing and securing sensitive data and adjusted protocols as needed.

Machine Learning

The method employed in this project is a Random Forest Classifier, a supervised machine learning algorithm designed to predict whether a credit card application will be approved or denied. It utilizes input features such as Income, Debt, Credit Score, and Prior Defaults to make decisions. Random Forest achieves this by generating multiple decision trees and combining their outputs to produce robust and accurate predictions.

The development of the Random Forest Classifier began with preprocessing the dataset to address missing values, normalize numeric fields, and encode categorical variables. The data was then divided into training and testing subsets, with 70% used for training and 30% reserved for testing, ensuring an unbiased evaluation of the model. Hyperparameters, including n\_estimators, max\_depth, and min\_samples\_split, were fine-tuned using GridSearchCV to optimize model performance.

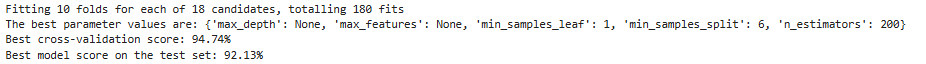
The Random Forest Classifier was selected for this project because it effectively handles both numerical and categorical data without requiring extensive preprocessing. Additionally, it provides insights into feature importance, which helps in making informed business decisions. During testing, Random Forest achieved a 92.13% accuracy on the test data. Its scalability also makes it a strong choice for managing the increasing number of credit card applications processed by Iron Financial Services.

Validation

Validation method and accuracy of model:

* Accuracy Score: The model achieved an accuracy score of 92.3%, representing the proportion of correctly predicted outcomes on the test set. This surpasses the target accuracy of 85%, demonstrating the model's reliability in predicting credit approval and denial outcomes.
* ROC-AUC Score: The Receiver Operating Characteristic - Area Under the Curve (ROC-AUC) score evaluates the model's ability to distinguish between approved and denied applications. The model achieved an AUC of 0.96 before hyperparameter tuning, which improved to 0.97 afterward, indicating excellent discriminatory power.
* Precision: Precision measures how many of the predicted approvals were actually correct. This metric is crucial to avoid approving applications that should have been denied, which could lead to financial losses. The model achieved a precision score of 0.93, reflecting strong accuracy in identifying legitimate approvals.
* Recall: Recall, at 0.91, measures the model’s ability to correctly identify all eligible applicants. This score ensures that the model minimizes the number of genuinely eligible applicants being denied.
* F1 Score: The F1 score, which balances precision and recall, was 0.92. This indicates a well-optimized model that maintains a good balance between precision and recall

The output below is the results of using GridSearchCV to optimize model performance:



A screenshot of a computer

Description automatically generated

Visualizations

A graph of credit card approval

Description automatically generated

A graph of credit card approval

Description automatically generated

A graph of credit card approval

Description automatically generated

A graph of credit card approval

Description automatically generated

A graph with blue and white bars

Description automatically generated

A yellow and purple squares with numbers and labels

Description automatically generated

A graph with numbers and a line

Description automatically generated with medium confidence

User Guide

Steps to Execute and Use the Application

1. Download Project Files:

* Jupyter notebook file- Credit\_Card\_Approval.ipynb
* CSV data- cc\_data\_randomized.csv
* Ensure these files are in the same directory.

1. Install Necessary Software and Libraries:

* Install Python 3.9 or higher.
* Install required libraries using the following command:
  + pip install jupyter pandas numpy scikit-learn matplotlib ipywidgets

1. Launch the Notebook:

* Open the Jupyter Notebook using the command: jupyter notebook
* The jupyter notebook window should open
* Locate the directory you saved them in and click on the Credit Card Approval.ipynb file
* Execute all cells sequentially to initialize the application using pressing ctl +enter
* Use the Interactive Widget:

The interactive widget will display input fields for the following attributes:

* Gender: Select "Male" or "Female" from the dropdown.
* Married: Select "Yes" or "No" from the dropdown.
* Bank Customer: Indicate if the applicant is a current bank customer.
* Prior Defaults: Specify if the applicant has defaulted on payments before.
* Employed: Indicate employment status.
* Drivers License: Indicate if the applicant has a valid driver's license.
* Citizen: Specify if the applicant is a U.S. citizen or not.
* Ethnicity: Select from "Asian," "Black," "Hispanic," "White," or "Other."
* Education Level: Choose from "PhD," "Graduate," "College," or "High School."
* Income: Enter the applicant’s annual income in dollars (e.g., 70,000).
* Debt: Enter the applicant’s total outstanding debt in dollars (e.g., 10,000).
* Years Employed: Enter the number of years the applicant has been employed.
* Credit Score: Enter the applicant's credit score (e.g., 750).

Note:

The Zip Code field is excluded as it is not required for the prediction. However, all other fields must be completed for the system to function accurately.

A screenshot of a computer

Description automatically generated

Steps:

Enter values into the interactive widget.

Press the Predict button.

View the system's output, e.g., "Result: Approved ✅", along with metrics explaining the decision

SOURCES: No sources used