Income Prediction for Nonprofit Donations

Student's Name

Western Governors University

Table of Contents

PROJECT OVERVIEW

A1. Research Question / Organizational Need

After randomly sending nearly 32,000 letters to community members, nonprofit organization Sunshine Benevolence Fund discovered that every donation received was contributed by individuals earning an annual income exceeding $50,000. The organization needs a method that can effectively identify potential donors while minimizing the cost of outreach. This project compares supervised machine learning algorithms to create an accurate model capable of predicting whether an individual earns more than $50,000. The project also seeks to accept or reject the null hypothesis that Sunshine Benevolence Fund cannot significantly improve its ability to efficiently reach potential donors by using predictive analytics.

A2. Project Scope

In this project, data preprocessing, exploratory data analysis, feature engineering, model training, and evaluation will be conducted using the Python programming language within a Jupyter Notebook environment. The dataset that will be used is a previously cleaned CSV file that was extracted from the 1994 U.S. Census data. The solution will include transforming features and splitting the data into training and testing sets. Machine learning models will be trained and evaluated using training and prediction times as well as accuracy and F-score metrics. The best performing model will then be optimized using grid search and cross-validation. Feature importance analysis will be conducted to assess the significance of different features. A baseline will be compared with the supervised learning algorithm, and a hypothesis test will be performed to determine if predictive analytics can significantly improve the charity's ability to efficiently reach potential donors when compared to their current approach. This will result in a reliable and deliverable predictive model for predicting individuals' income. Project stakeholders will all play a role in the success of this project. The organization will provide funding and support, donors will provide financial resources, executives will provide guidance and direction, fundraising teams will reach out to potential donors, and data analysts will analyze data to help make

decisions. There are no project constraints. The project scope does not include data collection or model deployment.

A3. Solution Overview: Tools and Methodologies

This project used the Python programming language inside a Jupyter Notebook environment which allowed me to view and test my code incrementally. I was able to document my findings throughout the project using markdown cells in the notebook. The only dataset used in the project was the census.csv file, which was read into a data frame using Pandas. Throughout the project, I utilized several analytical tools and methodologies, beginning with data preprocessing using libraries such as NumPy and Pandas. Exploratory data analysis techniques were applied, including summary statistics, log transformation for skewed features, and data visualization using the Seaborn and Matplotlib libraries. The data was preprocessed and engineered using techniques like scaling and one-hot encoding. Machine learning models were trained and evaluated using the train_predict function. Grid search with cross-validation was performed to optimize the chosen model using the GridSearchCV class from the scikit-learn library. Feature importances were computed and visualized, creating a reduced feature set. Hypothesis testing was conducted using a paired t-test. The ttest_rel function from the scipy.stats module was used to perform this test. The results of the test allowed me to reject the null hypothesis and accept the alternative that Sunshine Benevolence Fund can significantly improve its ability to efficiently reach potential donors by using predictive analytics.

PROJECT EXECUTION

B1. Project Plan

The hypothesis was reworded to add clarity and specificity. As a result, my method of hypothesis testing shifted from a one-sample t-test, which compared the performance of the predictive model to a baseline dummy classifier, to a paired t-test, which compared the performance of the predictive model to a naive

predictor. Despite this change, it did not have a significant impact on the project plan or outcome. All objectives were successfully achieved, methodologies were followed, and deliverables were produced as expected in the project plan and delivered early.

B2. Project Planning Methodology

As planned, the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology was used to manage this project. CRISP-DM provides a structured approach to planning a machine learning project. Specific sections of Python code corresponded to the different phases of the CRISP-DM methodology.

1. **Business understanding**
    - The specific needs of the charity as it related to this project were assessed through direct interaction and research.
    - Tools and technologies were assessed and gathered.
2. **Data understanding**
    - The necessary data was acquired and loaded.
    - Exploratory data analysis was conducted to understand the feature set and uncover patterns or trends. This provided insights into the dataset's characteristics and helped to identify key factors that could affect the project's success.
3. **Data preparation**
    - The data preprocessing section handled any missing values and transformed features to ensure the data was clean and ready for analysis.
    - Feature engineering was performed on the data to obtain or modify additional features.
4. **Modeling**
    - The model training section involved identifying which machine learning algorithms to use and developing models from these algorithms.
    - Technical assessment of these models occurred during this phase.

5. **Evaluation**
    - The model evaluation section involved evaluating machine learning models using accuracy and F-score performance metrics to determine the most appropriate model.
    - Training and prediction times were to be factored into the overall evaluation but were scrapped because there was only one candidate model that met the success criteria.
    - Accuracy and F-scores helped to perform quantitative evaluation of the chosen model.
    - Hypothesis testing showed that the chosen model was capable of significantly outperforming the charity's current approach. This enabled me to recommend the model to the charity.
6. **Deployment**
    - This phase was not applicable as it was outside of the project scope. However, it could have been used in the future if the charity's decision-makers chose to proceed with the implementation of the chosen model. At that point, a plan for deploying the model would have been developed and documented. Any maintenance or enhancements to the chosen model as well as subsequent project reviews and reports would also have happened in this phase.

B3. Project Timeline and Milestones

The project was completed ahead of schedule, with each milestone completed significantly faster than expected. The following table shows the planned and actual start and end dates for each milestone:

Milestone	Projected Start Date	Projected End Date	Actual Start Date	Actual End Date	Actual Duration (days/hours)
Understanding the charity's goals and assessing available data	6/14/2023	6/14/2023	6/14/2023	6/14/2023	~4 hours
Data collection & exploration	6/15/2023	6/17/2023	6/14/2023	6/14/2023	~2 hours
Preparing the data for modeling	6/18/2023	6/19/2023	6/15/2023	6/15/2023	~6 hours
Developing and training a machine learning model	6/20/2023	6/22/2023	6/16/2023	6/17/2023	2 days
Model performance evaluation	6/23/2023	6/23/2023	6/18/2023	6/18/2023	~2 hours

The timeline was adjusted largely due to no data collection being necessary. The existing CSV file was easily downloaded online and was clean and easy to work with. The dataset contained only fourteen features, which made data exploration and preparation much easier than anticipated. However, it still contained everything I needed to perform my analysis and achieve my objectives. My familiarity with Python, the libraries used, and Jupyter Notebooks also greatly contributed to the shortened timeline, particularly when developing and training the machine learning model.

METHODOLOGY

C1. Data Collection Process

- Actual Data Selection vs. Planned Collection Process:

  I correctly predicted that the data selection process would be straightforward because I had prior knowledge of the most appropriate dataset for this project. A quick web search led me to the GitHub repository "DSND_Term1" hosting the census.csv file. The file was easily downloaded to my personal computer and used throughout the project to analyze demographic trends. As expected, the preexisting dataset eliminated the need for data collection.

- Obstacles to Data Collection:

  I encountered no obstacles to data collection as data collection was outside of the project scope.

- Unplanned Data Governance Handling:

  I encountered no unplanned data governance handling. I anticipated the need to take precautions with the demographic data to avoid making any presumptions that could have influenced the project outcome. I took these precautions accordingly. The dataset used in this project is subject to the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To ensure I adhere to the terms of this license, I have appropriately attributed the original authors of the dataset, I have not made any permanent modifications to the dataset, and I have not engaged in any commercial activities involving the dataset.

C2. Advantages and Limitations of the Dataset

The dataset was clean and ready for analysis, so there were only a few limitations encountered in the project. These include:

- The dataset was created using data from the 1994 U.S. Census, making the data nearly thirty years old. The socioeconomic and demographic landscape have likely evolved since then. It

would have been ideal to use data from a more recent census, but the data from the older census was still sufficient for the purposes of this project.

- Demographic data such as the features found in the dataset inherently open the door to bias. However, precautions were taken to prevent bias from entering the project.
- The dataset lacks potentially important features that could influence an individual's likelihood to donate to charity, such as location, number of dependents, and occupational seniority.

These limitations did not have an impact on the outcome of the project, and were more than offset by the dataset's advantages, which include the following:

- It has features such as demographics, occupation, education, and employment that are all relevant to the project's goal of predicting income.
- It is free from data quality issues such as missing values or bad entries. This ensures accurate and reliable data is used when training the machine learning models.
- It has a sufficient number of records to ensure variability and diversity of potential donors.
- It is publicly available.

D1. Data Extraction and Preparation Processes

This dataset has been adapted from a previously published paper titled "Scaling Up the Accuracy of Naive-Bayes Classifiers: a Decision-Tree Hybrid" authored by Ron Kohavi (1996). The original dataset was extracted from the 1994 U.S. Census database. The adapted dataset specifically used in this project includes minor modifications to the original dataset. These modifications involve the removal of the 'fnlwgt' feature and the exclusion of records with missing or incorrectly formatted entries.

After the census.csv file was downloaded, I used Python's Pandas library to read the file into a data frame. I was then able to view the structure of the data, at which point I verified the dataset to be free of null values. Exploring the income data gave me a preliminary look at the statistical breakdown of

individuals making more and less than $50,000 annually. Next, I used Pandas to separate the target variable ‘income’ column from the feature set. During my exploration of the dataset, I noticed that two features, ‘capital-gain’ and ‘capital-loss,’ were likely skewed by outliers. To reduce the range of values caused by these outliers, I transformed those two features using the natural logarithm function from the NumPy library. I then imported the MinMaxScaler from the scikit-learn library to normalize all numerical features so that they had equal weight. This step was necessary to make sure that each feature was treated equally when applying the supervised machine learning algorithms. Categorical features were then converted to numerical values using one-hot encoding by way of Pandas’ get_dummies function. The target variable 'income' was converted into binary values, with 0 representing annual income less than or equal to $50,000 and 1 representing income over $50,000. The final step of the preparation process involved splitting the data into training and testing sets using the train_test_split function from the scikit-learn library.

E1. Data Analysis Methods

The goal of this project was to build and deliver a supervised machine learning model that could accurately predict whether an individual makes more than $50,000. I chose to train and test a total of three supervised machine learning algorithms on the dataset: Gaussian Naive Bayes, AdaBoost **c**lassifier, and logistic regression. I chose these algorithms because the dataset was clean and appropriately sized for all three. They were also suitable for the binary classification problem of whether an individual makes more than $50,000. At the outset of the project, I assumed logistic regression would be the best data analysis method to build the model. However, the AdaBoost classifier was actually the most accurate at predicting an individual's income level. Content with the outcome of the three-way test, I optimized the AdaBoost model and then did further analysis using full and reduced feature sets. I was able to do hypothesis testing comparing the performance of the optimized model and a naive model. The results of this analysis allowed me to reject the null hypothesis that Sunshine Benevolence Fund cannot significantly improve its ability to efficiently reach potential donors by using predictive analytics.

E2. Advantages and Limitations of Tools and Techniques

Entering the project, I had extensive familiarity using the Python programming language and its libraries in Jupyter Notebooks, so I chose to perform all data analysis using those tools within that environment. Python has a large set of libraries and packages that simplified data extraction, transformation, and analysis. Working in Jupyter Notebooks allowed me to run my code and instantly see the results. This proved invaluable as I was frequently able to evaluate and fix any errors in the code as the project progressed. I was able to use markdown cells in the notebook to document my findings and make notes of my thought processes. I prefer creating visualizations in Tableau since it has a richer feature set, but Jupyter Notebooks was perfectly capable of displaying the basic histograms and bar charts used in this project. I did experience some computational lag, particularly when performing grid search with cross-validation for hyperparameter tuning of the AdaBoost model. However, this lasted just a few minutes and was the only noticeable limitation of the tools used in the project.

Predictive analytics was the best technique to achieve the project goal. The AdaBoost classifier created an accurate and reliable model using predictive analysis. This model will provide data-driven decision making for the charity to help them improve marketing and operational efficiency. “AdaBoost is best used in a dataset with low noise, when computational complexity or timeliness of results is not a main concern and when there are not enough resources for broader hyperparameter tuning due to lack of time and knowledge of the user” (Santos, 2020). Compared to other algorithms, AdaBoost classifier is simpler to operate, requires fewer parameter adjustments, and is less prone to overfitting. A limitation of AdaBoost is that it is sensitive to noisy data which can cause the algorithm to focus too much on learning extreme cases. This can lead to skewed results and poor performance (Santos, 2020). The dataset used in this project was of high quality, however, so this was not an issue.

E3. Application of Analytical Methods

I applied the analytical methods mentioned above in the following steps:

1. **Data Preparation**:
    - I read the data from a CSV file and performed initial data exploration by displaying sample records, data information, and summary statistics.
    - I checked for null values in the dataset and printed the results.
2. **Data Preprocessing**:
    - I performed data preprocessing steps such as handling skewed features and scaling numerical features using logarithmic transformation and MinMaxScaler, respectively.
    - I one-hot encoded categorical features using Pandas' get_dummies function.
    - I encoded the target variable ('income') as 0 for '<=50K' and 1 for '>50K'.
3. **Train-Test Split**:
    - I split the preprocessed data into training and testing sets using the train_test_split function from sklearn.model_selection.
4. **Naive Predictor**:
    - I calculated the accuracy, recall, precision, and F-score for a naive predictor that always predicted the majority class ('<=50K').
5. **Model Training and Evaluation**:
    - I defined three classifiers: GaussianNB, AdaBoostClassifier, and LogisticRegression.
    - I defined the train_predict function to train and evaluate the classifiers with different sample sizes.
    - I trained and evaluated the classifiers on the training and testing data and stored the results in a dictionary.
    - I used Matplotlib to visualize the performance of the three classifiers in a series of grouped bar charts.

6. **Model Selection and Hyperparameter Tuning**:
    - I selected the AdaBoost classifier as the best model based on the F-score and accuracy on the testing data.
    - I used GridSearchCV for hyperparameter tuning of the AdaBoost model, performing grid search with cross-validation.
7. **Feature Importance**:
    - I computed the feature importances using the trained AdaBoost model and generated a bar chart of the five most import features.
8. **Model Evaluation with Reduced Features**:
    - I created a reduced feature set by selecting the top five important features.
    - I trained and evaluated the AdaBoost model on the reduced feature set.
    - I compared the performance metrics of the model trained on the full feature set to the performance metrics of the model trained on the reduced feature set to determine which version performed best.
9. **Hypothesis Testing**:
    - I conducted a paired t-test to determine if the optimized AdaBoost model significantly outperformed the naive predictor.
    - I printed the results of this test as well as the values of the t-statistic and p-value.
10. **Statistical Significance Visualization**:
    - I created a bar plot to display statistical significance by comparing the accuracy of the final model with the naive model.

I could not find a detailed description of the original data collection process, so I assumed that the dataset was a random sample of data from the 1994 U.S. Census and not limited to the charity's home state of California. Targeted sampling would likely introduce selection bias which would not provide an accurate

representation of the population. I also assumed that the features in this sample were conditionally independent, as is assumed by the Gaussian Naive Bayes algorithm. The results of the Naive Bayes model were considerably worse than the other two models, however, which may indicate dependent variables in the feature set.

RESULTS

F1. Statistical Significance

Once the best hyperparameters for the AdaBoost classifier were determined, I compared the performance metrics of the unoptimized model with the optimized model as seen in the following table:

Metric	Unoptimized Model	Optimized Model
Accuracy Score	0.858	0.868
F-score	0.725	0.746

The optimized model's accuracy on the testing data was 0.868, while its F-score was 0.746. These scores were slightly better than those of the unoptimized model's accuracy and F-score values of 0.858 and 0.725, respectively. However, compared to the previously found naive predictor accuracy (0.248) and F-score (0.292) benchmarks, the results of the optimized model were a significant improvement. I then wanted to see if I should use the reduced feature set to train the final model. When using the reduced feature set, the optimized model's accuracy and F-scores were both lower than the same scores when using the full feature set. The testing data accuracy went from 0.868 to 0.842, a 3% decrease. The testing data F-score difference was more noticeable, dropping from 0.746 to 0.699. This was a 6.3% decrease in score. I determined that while the accuracy score of using the reduced feature set met the criteria for success that was previously set at 0.80, the F-score fell short of the target score of 0.75. The reduction in

training time made possible by the reduced feature set was irrelevant because the criteria for project success was not met.

To test the hypothesis, I calculated the t-statistic and p-value using a paired t-test. The ttest_rel function performed the test, comparing the accuracy scores of the final AdaBoost model and the naive predictor. The naive predictor accuracy was calculated to be 0.244 while the final model accuracy was 0.868, a significant improvement. I set the significance level to alpha = 0.05. The t-statistic (-190) and sub-alpha p-value (0) were strong evidence against the null hypothesis. I was therefore able to reject the null hypothesis and conclude that the final model significantly outperformed the charity's current approach.

F2. Practical Significance

The AdaBoost model will provide a high level of practical significance and value for the charity's decision-makers. The model's ability to accurately predict whether an individual makes over $50,000 annually will help the charity target potential donors more effectively via marketing strategies such as fundraising campaigns and customized cold calls. This targeted approach will decrease the money wasted on random solicitation such as the 32,000 letters previously mailed by the organization. The organization can implement the model however they choose and will be able to apply any up-to-date demographic data. The results of this project will provide valuable donor information such as demographics and preferences, increasing the ability to develop lasting relationships with donors to encourage long-term donations. The charity will also have the knowledge that capital gains and losses, age, hours worked per week, and education level are the most important factors that can be used to identify potential donors. Knowing whom to request donations from will allow the charity to use organizational resources more efficiently and increase the decision makers' ability to strategically plan. This will all contribute to Sunshine Benevolence Fund seeing an increase in donations and a decrease in both time and resources.

F3. Overall Success

The goal of this project was to construct a supervised machine learning model that accurately predicts whether an individual makes more than $50,000. Overall, this project was successful in achieving this goal. All project objectives were met with few challenges, and all project milestones were completed either on time or early. The following table illustrates the minimum scores for evaluating the success of project execution compared to the actual scores that were a result of the performance evaluations:

Metric	Successful Score	Actual Score
Accuracy Score	0.80	0.87
F-score	0.75	0.75
P-value	< 0.05	0

All three criteria for measuring the success and effectiveness of the project outcomes were satisfied, which indicates a successful project.

G1. Summary of Conclusions

As a result of my analysis, I was able to deliver a model that could be used to predict individual income levels with a high degree of accuracy. Sunshine Benevolence Fund will greatly benefit from using this model to identify potential donors. The charity had been randomly sending letters to solicit donations, wasting precious time, resources, and money. Once they discovered that only individuals making over $50,000 per year were donating to them, they decided to act. I calculated that only 25% of individuals in the dataset made more than this amount, indicating that the organization was making the right decision in trying to stop the wasteful practice of mailing random letters. The naive predictor benchmarks of 25% accuracy and 29% F-score are consistent with this income analysis, so I made that accuracy score the baseline. The optimized model's accuracy on the testing data was 87%, while its F-score was 75%, a significant improvement. The results of a paired t-test allowed me to easily reject the null hypothesis.

After developing and evaluating the final model, I can confidently say that its implementation can help the charity become significantly more efficient at reaching potential donors.

G2. Effective Storytelling

Effective storytelling in a project such as this is key. It enhances comprehension, provides clarity, and influences decision-making. This is essential for a successful project outcome. Thankfully, I was able to use effective storytelling throughout the project. By using Jupyter Notebooks, I was able to seamlessly combine code cells with markdown cells. After writing a section of Python code, I executed the code and immediately displayed any output that resulted. Markdown cells were used throughout the data analysis process to title each section of code, document my findings, and provide insights. I used string formatting for added clarity when outputting quantitative values ("Training set has 36,177 samples. Testing set has 9,045 samples.") I chose meaningful variable names so that anyone looking at the code would more easily understand the data or operations being used. Histograms aided me in data preprocessing, and I even relied solely on a series of grouped bar charts to select the best performing model among the three that were tested. Additionally, another bar chart displayed the top five important dataset features for determining income. I concluded the code by presenting a bar chart that illustrates the disparity between the accuracy scores of the naive model and the optimized AdaBoost model. These visualizations not only aided me in the analysis process but also serve to assist other project stakeholders in agreeing with my findings.

G3. Findings-based Recommendations

Based on the findings, my first recommendation to Sunshine Benevolence Fund would be that they implement a predictive analytics system to identify potential donors more accurately and efficiently. The system would use the optimized AdaBoost model, which was developed as part of this project. This

targeted approach would improve fundraising campaigns and ensure that the charity's resources are properly distributed to boost donor engagement and support.

I would also recommend that the charity modify its donor outreach strategy to engage potential donors in a more personalized and targeted manner. The charity can do this by considering the five most important demographic features identified in the analysis. Regular evaluation and modification of the outreach strategy will help the charity stay adaptive and responsive to donor preferences.

H. Panopto Presentation

Panopto presentation link: https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=c0ba056a-9b6c-472b-8272-b02f014b7736

APPENDICES

I. Evidence of Completion

The following files have been included in my submission and provide evidence that this project has been completed:

1. Jupyter Notebook ipynb file displaying all Python code used in the project (“Capstone.ipynb”)
2. Dataset as comma-separated values file (“census.csv”; UDACITY, 2021)
3. Microsoft PowerPoint Presentation file (“Capstone-Presentation.pptx”)

SOURCES

Kohavi, R. (1996). Census Income. UCI Machine Learning Repository. Retrieved from: https://doi.org/10.24432/C5GP7S

Santos, G. (2020). The Ultimate Guide to AdaBoost, Random Forests and XGBoost. Towards Data Science. Retrieved from https://towardsdatascience.com/the-ultimate-guide-to-adaboost-random-forests-and-xgboost-7f9327061c4f

UDACITY. (2021). CharityML Project Repository. GitHub. Retrieved from: github.com/udacity/DSND_Term1/tree/master/projects/p1_charityml