March 18, 2024

C964: Computer Science Capstone

[Part A: Letter of Transmittal](#_heading=h.gjdgxs) 2

[Letter of Transmittal Requirements 2](#_heading=h.2jxsxqh)

[Letter Template 2](#_heading=h.z337ya)

[Part B: Project Proposal Plan](#_heading=h.30j0zll) 4

[Project Summary](#_heading=h.1fob9te) 4

[Data Summary](#_heading=h.3znysh7) 5

[Implementation](#_heading=h.2et92p0) 5

[Timeline](#_heading=h.tyjcwt) 7

[Evaluation Plan](#_heading=h.3dy6vkm) 8

[Resources and Costs](#_heading=h.1t3h5sf) 9

[Part C: Application](#_heading=h.4d34og8) 10

[Part D: Post-implementation Report](#_heading=h.2s8eyo1) 10

[Solution Summary](#_heading=h.17dp8vu) 10

[Data Summary](#_heading=h.3rdcrjn) 10

[Machine Learning](#_heading=h.26in1rg) 13

[Validation](#_heading=h.lnxbz9) 13

[Visualizations](#_heading=h.35nkun2) 14

Lessons Learned…………………………………………………………………………………………………………………………………16

[User Guide](#_heading=h.1ksv4uv) 17

# Part A: Letter of Transmittal

March 18, 2024

Mr. Tom Morrow

Valley Kestrel Insurance

645 Hualapai Highway 89, Peoria, AZ, 85345

Dear Tom Morrow,

Good morning, I hope you are doing well. I am getting back to you regarding your suggestion for the company’s improvement. Valley Kestrel works tirelessly to provide the best vehicle insurance nationwide, and at an affordable price. I understand that due to the increasing inflation and cost of doing business, the company has had to make the unfortunate decision to increase premiums for customers starting in 2024. To help alleviate the financial blow to customers, the company has decided to perform marginal increases to the premiums, rather than place the burden of increasing costs solely on customers. Valley Kestrel also wants to keep the prices of premiums steady for the customer, dictated by safer driving, instead of having them fluctuate with the market.

As we previously discussed, I am proposing a potential solution to the situation Valley Kestrel finds itself in. Due to the recent crime that has occurred, vehicle thefts and break-ins continue to be a concern. With every incident, premiums will continue to increase more than expected. Our solution includes creating a machine learning model that can predict the likelihood of a theft or break-in depending on the make, model, and year of the customer’s vehicle, as well as the state the vehicle resides in. We have information on all 50 states as well as Washington D.C.

This solution will serve to inform customers on whether their vehicle is a high-value target for theft or vandalism so that they may take the necessary precautions and ultimately, keep the cost of their premiums low.

The costs of building, installing, and maintaining the model will cost upwards of $100,000 for the bare-bones implementation, with foreseeable operation for the next 5 to 10 years. This total includes labor, location of the server which will house the model, and the server itself. The model is expected to be completed within three months. Testing will be expected to last for one week afterwards. Customer data will be kept confidential in accordance with data privacy laws Data integrity will also be upheld, as well as appropriate access to the data.

I hold a BS in Computer Science and a MS in Statistics. I have worked in the field of statistical analysis for the past 6 years and also worked with our team of data scientists and engineers for the past 3 years to develop machine learning models for various purposes. I want to reassure you that the health and safety of Valley Kestrel is our top priority. I just ask that when the model is ready for the launch, that you keep your customers informed about it. If at any point you require additional assistance, a team of customer service representatives will be ready to take any questions.

Sincerely,

[Redacted Signature]  
[Redacted], Customer Service Liaison

**Part B: Project Proposal Plan**

## Project Summary

Vehicle theft is among the most concerning issues facing customers and insurance providers today. With every theft, the insurance company must raise premiums to cover the cost of insuring vehicles. This depletes the customers’ funds further and makes them likely to switch providers to save money.   
  
Valley Kestrel Insurance aims to provide the best possible coverage while providing incentives for safe driving and the lowest cost to the customer. As a newer provider, Valley Kestrel cannot afford to lose its customer base to the ongoing theft. In an effort to mitigate this, our company has provided a possible solution to the problem.  
  
Upon completion of development, our company will seek to roll out a brand new application to Valley Kestrel’s customers. This app will use machine learning, a technique that allows computers themselves to “learn” from a set of data and draw conclusions based on the data. Our model (or app) will use an algorithm for predicting the chances that an event will occur. In our case, it will be able to predict the likelihood of vehicle theft depending on the make and model, the year, and state in which the vehicle is located. The initial version of this application will be available on mobile for Android and iOS, and a desktop version will be available shortly afterwards.   
  
This ML model aims to provide customers insight into the likelihood that particular vehicles will be stolen. This will allow them to take the necessary precautions when traveling throughout the US. This will in turn provide repeat business with existing customers by providing lower premiums than the competition and a useful tool to protect customers with. This also serves to attract new customers through word of mouth and advertising.

## 

## Data Summary

The dataset that will be used is available on Kaggle at <https://www.kaggle.com/datasets/thedevastator/uncovering-state-by-state-car-theft-trends-in-20>.

The dataset has been modified for the purposes of building the model. The dataset is collected into a .csv file and opened in Microsoft Excel for the purpose of preparing the data to train the model. The data has undergone preprocessing (cleaning) by removing the Rank column from the dataset. The other categorical variables (State, Make/Model) have been encoded through Label Encoding to convert them to numerical values to make it easier for the model to work with. The data collected from the model will be analyzed and presented on the Google Colab platform for viewing.  
  
This model will provide an interactive app for customers to use. This app will present the model’s findings and implement various visuals (graphs). This is important as it shows customers which vehicles are likely to be stolen or broken into.  
  
Customer data will be kept confidential according to data privacy laws. The data will remain unchanged and only authorized users will have access to it.

## Implementation

Our company believes in putting the customer first, so we prefer to use the Agile methodology in our development, unless otherwise instructed. Agile will allow us to collaborate more with Valley Kestrel and let the company have a more hands-on approach in the app’s development. Agile methodology also allows for fast sprints and welcomes change from the customer. Each phase of the Agile process is preceded by daily stand-up meetings and followed by sessions in which team members listen to and take notes on feedback given by the client. Retrospective meetings are also a part of this.

Here is a rough outline of development:

* **Planning:**   
   - A product roadmap is made and will serve to address the long-term goal of the project and direction for all team members involved  
   - Data analysts and engineers will use statistical research to inform them on how to best approach planning for the app’s development  
   - Team members will understand their roles and the work they are responsible for  
   - Resources will be allocated according to collaborative brainstorming; determining what tools and frameworks are best suited for creating the app are discussed
* **Design:**  
   - The engineering team will meet with the client (i.e. Valley Kestrel) to discuss their needs and what they expect to see on the finished app  
   - Different design concepts are reviewed; Documentation is drafted and updated as design work is further underway  
   - UML diagrams and a prototype app will be produced and released within a sandbox environment along with performance of unit testing
* **Development:** - The application is developed in-line which the process of Continuous Development and Integration - Test Driven Development is implemented for automated testing on the app   
   - Pair Programming is emphasized to speed up the code writing process and minimize the introduction of bugs into the code
* **Testing:**  
   - Higher-end testing is done on the application; this include functional, module, system, and regression testing  
   - White-box testing is done to check the code, Black-box testing is done in conjunction with the client to see if the app meets their needs or expectations   
   - Bug tracking and reporting is put into place
* **Deployment:**  
   - The team addresses rollback and rollforward strategies and put them into place  
   - The app is deployed in a beta state; continuous testing will still be in effect  
   - Bugs and glitches are reported and documented for repair and fine-tuning  
   - Once client is satisfied with the end result, final sign-off for the product is done and complete deployment takes place  
   - Post-deployment monitoring and support begins
* **Review:**  
   - All client feedback is collected and archived  
   - Lessons Learned: The team comes together to discussed the entire process; what worked, what could have been done better, and what lessons or takeaways could be used for the next project

**Timeline**

The days listed in the timeline include only work days.

| Milestone or deliverable | Duration  (hours or days) | Projected start date | Anticipated end date |
| --- | --- | --- | --- |
| Planning phase | 14 days | 3/20/2024 | 4/8/2024 |
| Design phase | 14 days | 4/9/2024 | 4/26/2024 |
| Development phase | 14 days | 4/29/2024 | 5/16/2024 |
| Testing phase | 14 days | 5/20/2023 | 6/6/2024 |
| Deployment phase  Deliverable on: 5/27/2024 | 4 days | 6/7/2023 | 6/12/2024 |
| Review phase | 2 days | 6/13/2024 | 6/14/2024 |
| **Total** | **62 days ≅ 3 months** |  |  |

## 

## Evaluation Plan Verification methods for each stage of development will be presented below:

* **Planning:**   
   - A product roadmap must be created   
   - Resources must be allocated properly
* **Design:**  
   - Design team must meet with the client at least once  
   - Documentation and UML diagrams drafted  
   - Prototype app created and released to sandbox environment
* **Development:**  
   - App created and continuously tested (CD/CI)  
   - Automated testing established  
   - Unit testing performed at least once
* **Testing:**  
   - White-box and Black-box testing done  
   - Bug tracking and reporting put in place
* **Deployed:**  
   - App deployed in beta  
   - Client sign-off  
   - Start of post-deployment monitoring
* **Review:**  
   - Lessons Learned: begin lessons learned retrospective meeting(s)  
   - Notes, documentation, and client feedback collected and archived for future reference

Validation method upon completion of the project includes the following:

* User Acceptance Testing is performed
* App is deployed in beta and tested by a select group of customers
* Client signs off on project

## 

## Resources and Costs

These represent a one-time cost for the app’s development:

| **Resource** | **Description** | **Cost** |
| --- | --- | --- |
| Automated Testing Tool Bundle | Bundle that includes various tools for automated testing for web, mobile, and desktop apps | $2,710 |
| Gaming PC | PC contains an Intel i9 CPU and GeForce RTX 4080 GPU to develop model on and for future proofing purposes | $4,230 |
| Data Engineer | The engineering team will perform research and use it to build the model | $110/hr \* 4 (Engineers) for Design and Development phases (22 days) |
| Software Developer | Developers that will work to build the app | $88/hr \* 2 (Developers) for Development, Testing, and Deployment phases (26 days) |
| On-Premises Server | server to store ML program and associated software | $100,000 |
| **Total** |  | $220,988 |

# These represent recurring costs for the app’s maintenance for a 5 year period:

| **Resource** | **Description** | **Cost** |
| --- | --- | --- |
| Power consumption | The server’s power consumption, measured in kiloWatts/hr | 0.25 kWh/hr \* 168 (hours in a week) \* 260 (weeks in 5 year period) |
| Software developer | Will work on maintenance of the app on top of regular duties | $88/hr \* 40 (hours a week) \* 47 (weeks out of the year that the SD works) \* (5 year period) |
| **Total** |  | $838,120 |

# 

# 

# Adding up the total from both one-time and recurring costs results in:

| One-time costs | $220,988 |
| --- | --- |
| Recurring costs | $838,120 |
| **Grand Total** | **$1,059,108** |

# 

# Part C: Application

1. The model uses a supervised algorithm, specifically a Decision Tree Classifier.
2. User interface is included in the app.

# Part D: Post-implementation Report

## Solution Summary

* Due to adverse economic conditions and heightened crime throughout the US, insurance providers were forced to increase premiums. This has put customers in an even difficult position by having to pay more when the price of all other goods and services has increased as well. Valley Kestrel’s aim was to provide fixed price premiums for customers without recent recorded incidents through the use of a custom built app.
* The application in question used an ML model to predict the likelihood that a certain vehicle would be the target of theft or vandalism. The customer would use this app to inform them if their own vehicle was a high value target so they could take the necessary measures to secure their vehicles against theft.

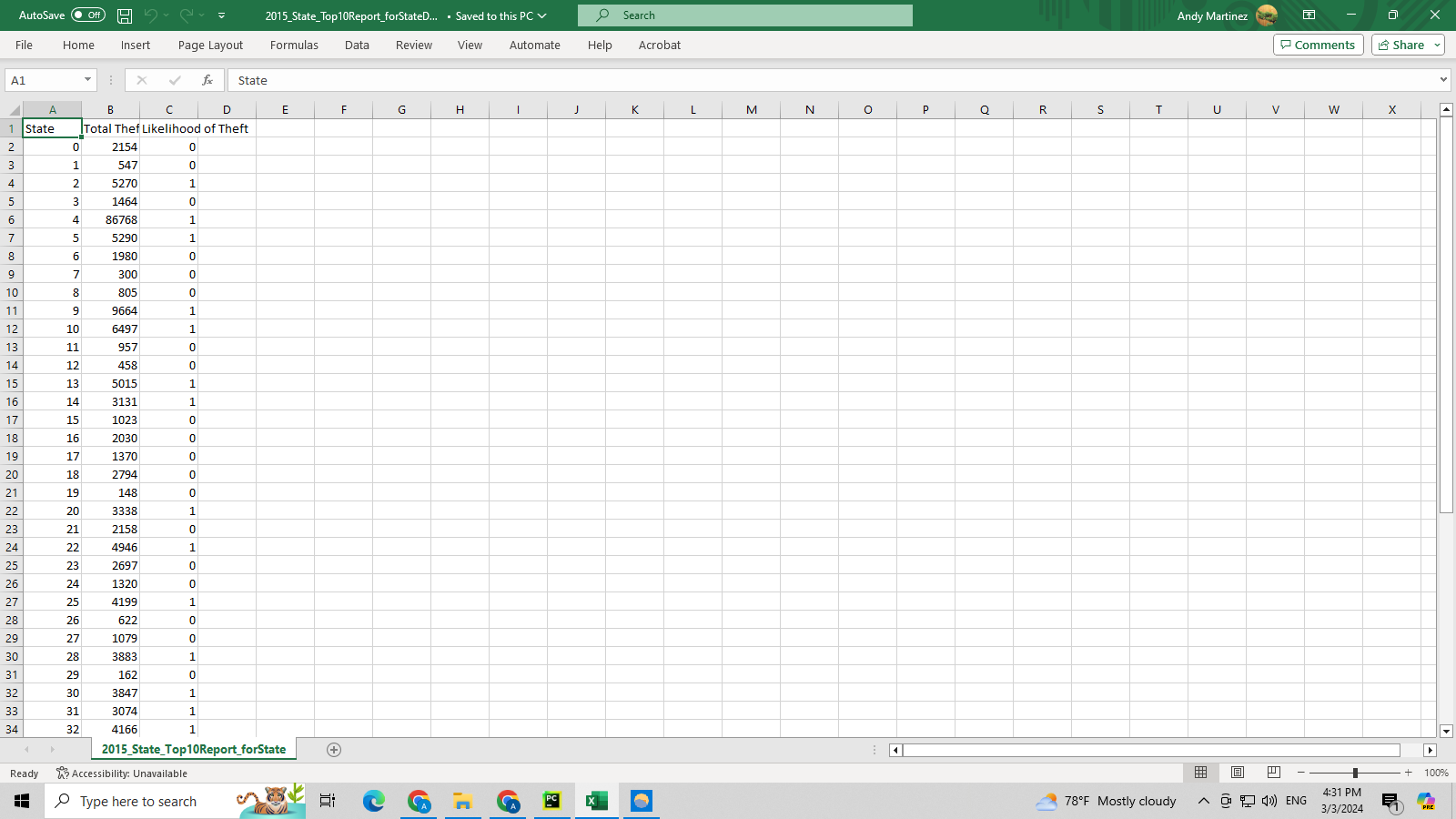
Data Summary

* The original dataset that was used was found on Kaggle's website and provided by Joe Boutros at this link: <https://www.kaggle.com/datasets/thedevastator/uncovering-state-by-state-car-theft-trends-in-20>.
* The original dataset contained six columns: Index, State, Rank, Make/Model, Model Year, and Thefts. Preprocessing the data involved creating four different subset datasets from the original, shown below:

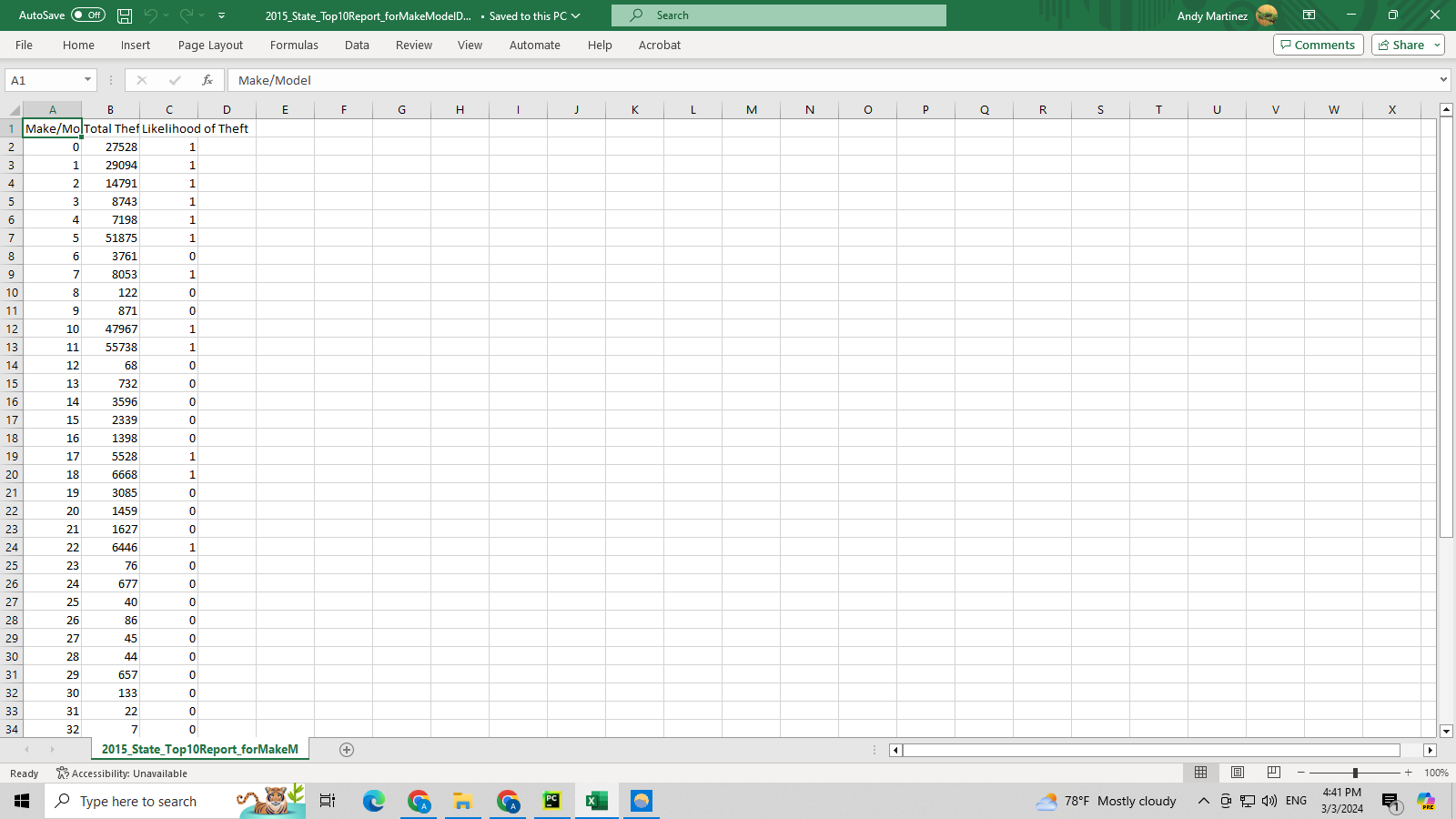
- 2015\_State\_Top10Report\_wTotalThefts.csv  
- 2015\_State\_Top10Report\_wTotalTheftsandModelYear.csv  
- 2015\_State\_Top10Report\_forStateDF.csv

- 2015\_State\_Top10Report\_forMakeModelDF.csv

* In all four datasets, the Index and Rank columns were deleted as the information contained in these two columns was deemed superfluous. The first dataset (without Model Year) was used to create both bar graphs. The second dataset with Model Year was used to create the pie chart.
* The third and fourth datasets have a new column, Likelihood of Theft, created using an Excel formula during preprocessing. In the third dataset, if the number of thefts per state was greater than 3500, then the number 1 was inserted into the cell. Otherwise, 0 was inserted instead. The same was done in the fourth dataset, except based on a different condition. If the number of thefts per vehicle was greater than 5000, then the number 1 was inserted into the cell. Otherwise, 0 was inserted instead.
* Preprocessing for the model datasets involved changing all string values to numerical values. The datasets had to be cut down even further by eliminating the Thefts column in both sets. The plan was to have just a singular X column to go with the singular Y column. This made it easier to build the model.









## 

## 

## 

## 

## 

## 

## Machine Learning

* The ML model uses a supervised algorithm known as Decision Tree Classifier. This algorithm uses a group of rules to make decisions based on input values.
* The model was developed in PyCharm 2023.2.3 (Community Edition) with creating a predicting model in mind. The prediction was done using user input (state and make and model of vehicle).
* This algorithm was chosen because it is particularly useful for solving binary classification problems in machine learning. The model was developed in an Integrated Development Environment as this gave the team the most flexibility in creating the model. Security was also a concern, thus the IDE was downloaded and set up on the gaming PC, providing an offline environment to develop the model in.

Validation

* The model was trained using 70% of the data and tested using the remaining 30%. Running an accuracy check on the state model resulted in an initial score of 0.625, while the score for the second model resulted in a score of 0.66.
* These results indicate that the models were fairly accurate, which the team was satisfied with, seeing as these were the initial releases of the models.. With future modifications and testing, the models can become more accurate.

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

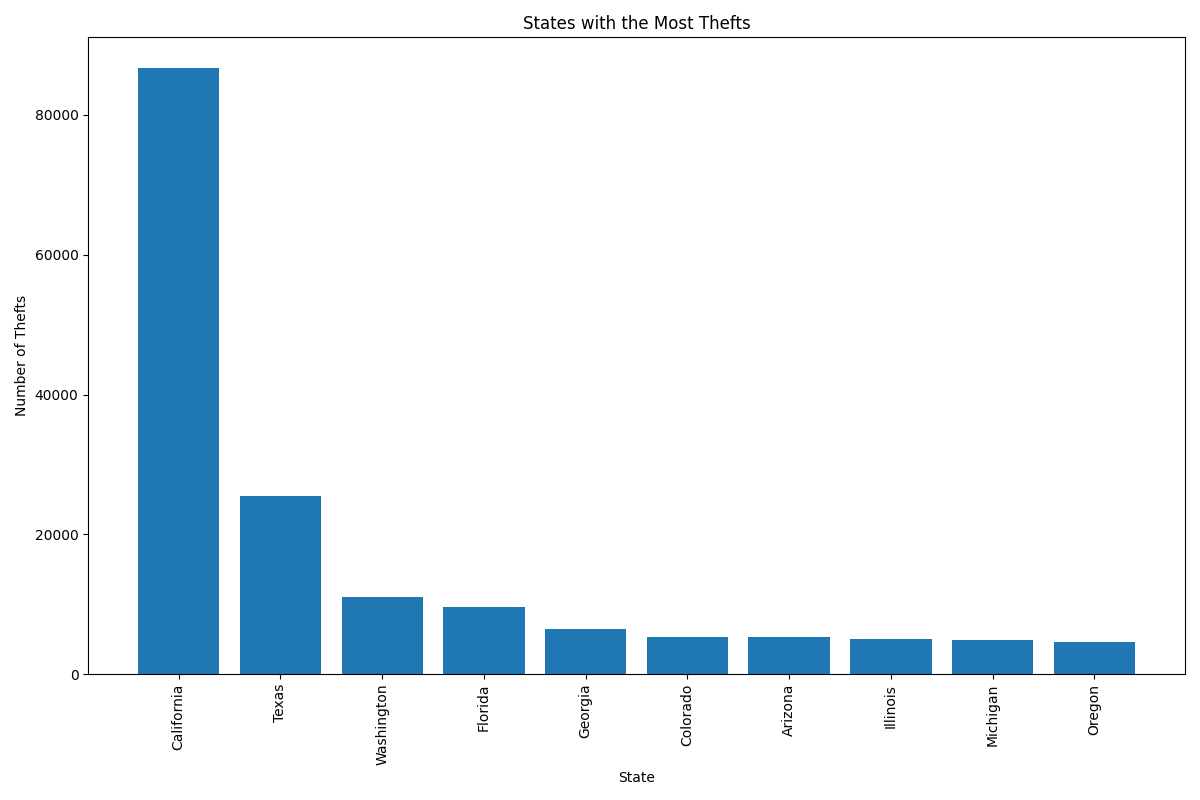
## 

## 

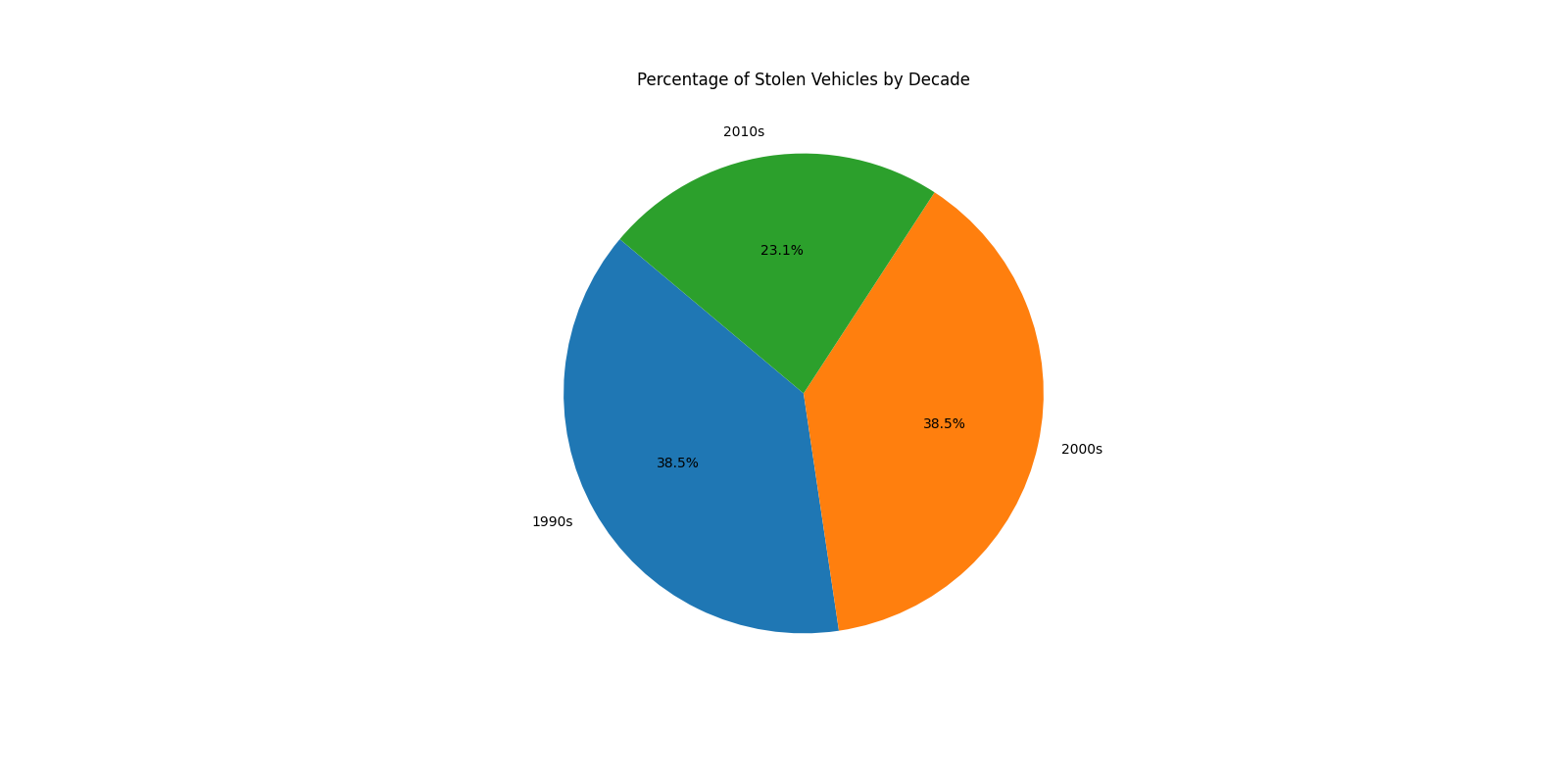
## Visualizations

## The three visualizations were generated using two datasets in the Pycharm Community Edition 2023.2.3. They are located in visuals.py and main.py source code files and are also included here:

This bar graph shows which vehicles are the most targeted through the US, based on the make and model. Sedans are the most common body types targeted, along with full size pickups.



This next bar graph shows which states have seen the most thefts.



Using a slightly different dataset, our team put together this pie chart showing which percentages of vehicles have been stolen the most. In the dataset, the oldest model year falls in the 1990s, while the newest model year is 2015. There is an even split at 38.5% for vehicles made in the 1990s and vehicles made in the 2000s, while newer vehicles made in the 2010s saw less theft at 23%.

Lessons Learned

There are a few conclusions that our team came to concerning vehicle theft. According to the research, Honda, Toyota, Chevrolet, and Dodge appear to be the makers of vehicles with the most theft. In addition, the top ten states found to be hot zones for theft include those with the highest populations. The high population can be attributed to many metropolitan areas across the US being found within these states. With higher population densities come greater opportunities for vehicle theft. Finally, the team found that vehicles produced in the 1990s and 2000s made up a higher percentage of total vehicles stolen than those made in the 2010s. One theory suggests that newer vehicle models are equipped with more anti-theft measures as opposed to older vehicles. We expect our findings to aid in keeping customers informed and will allow them to take the necessary measures to counter theft. Our company recommends customers invest in wheel locks, alarm systems, and tracking devices. Customers in major metropolitan areas should take extra precautions, such as parking in well-lit areas and keeping valuables out of sight.   
 Looking back, the model is rather simplistic in design and execution. Our proposal for improving the model would be to gather more information for our datasets. The vehicle data we had registered in our datasets stopped in 2015. Many vehicles were left off the dataset, such as those made by Suzuki or even sports cars. Seeing as this model was built almost ten years later, it is safe to say that many more types of vehicles have come out since then, including those manufactured by Tesla, a recent brand that has gained widespread popularity amongst consumers. With upcoming releases of the app, our team will be able to include more vehicles for the datasets.

## User Guide

The following is a user guide for installing and running the app on a single device. Note that each device varies in what specifications, operating systems, and hardware it runs on. This particular app was created and tested on a Windows 10 PC. Please adjust accordingly to ensure the app installs and runs properly on your workstation.

1. On your preferred browser, navigate to <https://www.jetbrains.com/pycharm/>.
2. Click the ‘Download’ button that appears and scroll down until you reach the Pycharm Community Edition. This is the free version of the IDE and is the preferred version to work with. Click the button to download.
3. Have the installer run and install the necessary files. Once finished, open Pycharm.
4. The zip file containing the app should have been downloaded. Unzip the file if you have not done so already.
5. Back in Pycharm, click on ‘Open’ and locate the folder called ‘CapstoneWGU964’ from the unzipped file. This is the project folder, and contains all the necessary files for the app to run. Select this folder and click ‘OK’.
6. In the ‘Main Menu’ (top ribbon), navigate to ‘File’, then ‘Settings’. Under ‘Project: CapstoneWGU964’, click on ‘Python Interpreter’. This is where the additional libraries will be installed. Hit the + sign and type in the following into the search bar: ‘scikit-learn’ and then hit ‘Install Package’. Wait until the package is installed (a green message will confirm this). Repeat this search for ‘matplotlib’, ‘scikit-learn-tree’, and ‘pandas’. Click ‘OK’ to exit this menu.
7. Everything should be prepared to run the app now. Ensure that main.py is the selected file to run and then click ‘Run’ (green replay button).