Fraud Prediction using AutoAI



*ARIN 5303 Week 4*

# Preface

## Overview

Automation and artificial intelligence (AI) are transforming businesses and will contribute to economic growth via contributions to productivity. They will also help address challenges in areas of healthcare, technology & other areas. At the same time, these technologies will transform the nature of work and the workplace itself. In this code pattern, we will focus on building state of the art systems for churning out predictions which can be used in different scenarios. We will try to predict fraudulent transactions which we know can reduce monetary loss and risk mitigation. The same approach can be used for predicting customer churn, demand and supply forecast and others. Building predictive models require time, effort and good knowledge of algorithms to create effective systems which can predict the outcome accurately. With that being said, IBM has introduced AutoAI which will automate all the tasks involved in building predictive models for different requirements. We will get to see how AutoAI can churn out great models quickly which will save time and effort and aid in faster decision-making process.

## Industry Use-case

### **A. Fraud detection in the insurance business**

Headquartered in Aurora, just outside of Chicago, Northeast Insurance Company, (NIC) employs over 5000 people across the continental United States.

During its fifty-year history, NIC has been struggling to detecting potentially fraudulent activity and has turned to IBM for their data science and AI offerings to predict fraud with insurance claims, before the claim is settled.

### **B. Business challenge story**

The global Fraud Detection and Prevention (FDP) market size is expected to grow from USD 20 billion to 63.5 billion by 2023, according to various analyst reports (i.e. “[Fraud Detection and Prevention Market by solution](https://www.marketsandmarkets.com/Market-Reports/fraud-detection-prevention-market-1312.html)). Predictive analytics segment is projected to be the largest contributor to the FDP market during the forecast period.

Predictive analytics solutions help enterprises identify the possibilities of fraud incidents by analyzing the current data. The solutions are used to identify potential threats, payment frauds, frauds in insurance processes, and credit/debit card frauds. Organizations are trying to impart these solutions for predicting fraud or suspicious activity and their pattern to help drastically reduce losses due to frauds.

Fraud analytics solutions employ sophisticated analytics and predictive modeling to identify potential fraud in real time during data entry, rather than during a later batch run after a transaction is complete. It can be applied to claims and underwriting fraud. Regional Tier 2 and 3 insurers are more likely to adopt SaaS-based point solutions for fraud analytics use cases. Larger insurers are implementing these solutions via professional services providers and system integrators.

Drivers usually sign a six-month policy with an auto insurance policy. Each month, or all at once, the driver pays a fee, or **premium**, to the company. There are a few things that determine the cost of the policy: the type of car insured (particularly its safety record and how expensive it is to repair) the driver's record (the more speeding tickets the driver has incurred, the riskier he is) and even age (teenagers cost more to insure because they're less experienced drivers, and therefore a bigger risk.) Lower-cost premiums are enjoyed by drivers with fewer accidents and tickets on their records, part-time drivers, people who take driver education courses, and families with multiple cars.



PAIN POINTS

Information siloed, overload, difficult to see clearly

Using AI/ML for fraud detection is not new. However, typical organization contains multiple fraud departments, each with its own internal point-solution which monitors fraud for that specific channel, product, or fraud type. Structured and unstructured data collected internally and externally but very few of these point-solutions share data. Each uses varying analytical techniques across channels and transaction systems, which results in not having a complete view of risk exposures across the institution. Cannot see patterns or behaviors that would spark a concern that fraudulent activity is crossing multi-business lines because the observation space is too narrow.

Difficult to predict fraud

Rare occurrences create an imbalance in the classification of fraud detection models and makes detection challenging.

Shift to increased digital and mobile customer platforms led to transactions being executed more quickly, leaving banks and processors with less time to identify, counteract, and recover the underlying funds. As quickly as new technology is used to identify fraudsters, they themselves are identifying new ways of defrauding the bank. For instance, identity theft is mutating from card skimming to account takeovers (ATO). Synthetic identify, a scenario where fraudsters combine fragments of stolen or fake information to create a new identity and apply for financial products.

Cost of (near) real-time detection is high

Organizations need to identify anomalies accurately and efficiently at the level of accounts, merchants, cardholders and locations.

False positives require manual investigations through providing content analytics across primary internal and external data sources

Fraud detection – meaning detecting fraudulent behavior after it occurs – forcing companies to set aside money and resources for the inevitable losses they will incur, costing financial institutions millions of dollars and destroying the customer experience. Financial institutions need to get in front of the problem and focus on fraud prevention.

72%

cite fraud as a growing concern over the past 12 months and nearly **63%**report

the same or higher levels of fraudulent losses over that

same period

$44B

Worldwide losses due to fraud by 2025

25%

of declined sales transactions for e-commerce merchants were false positives.

Tools

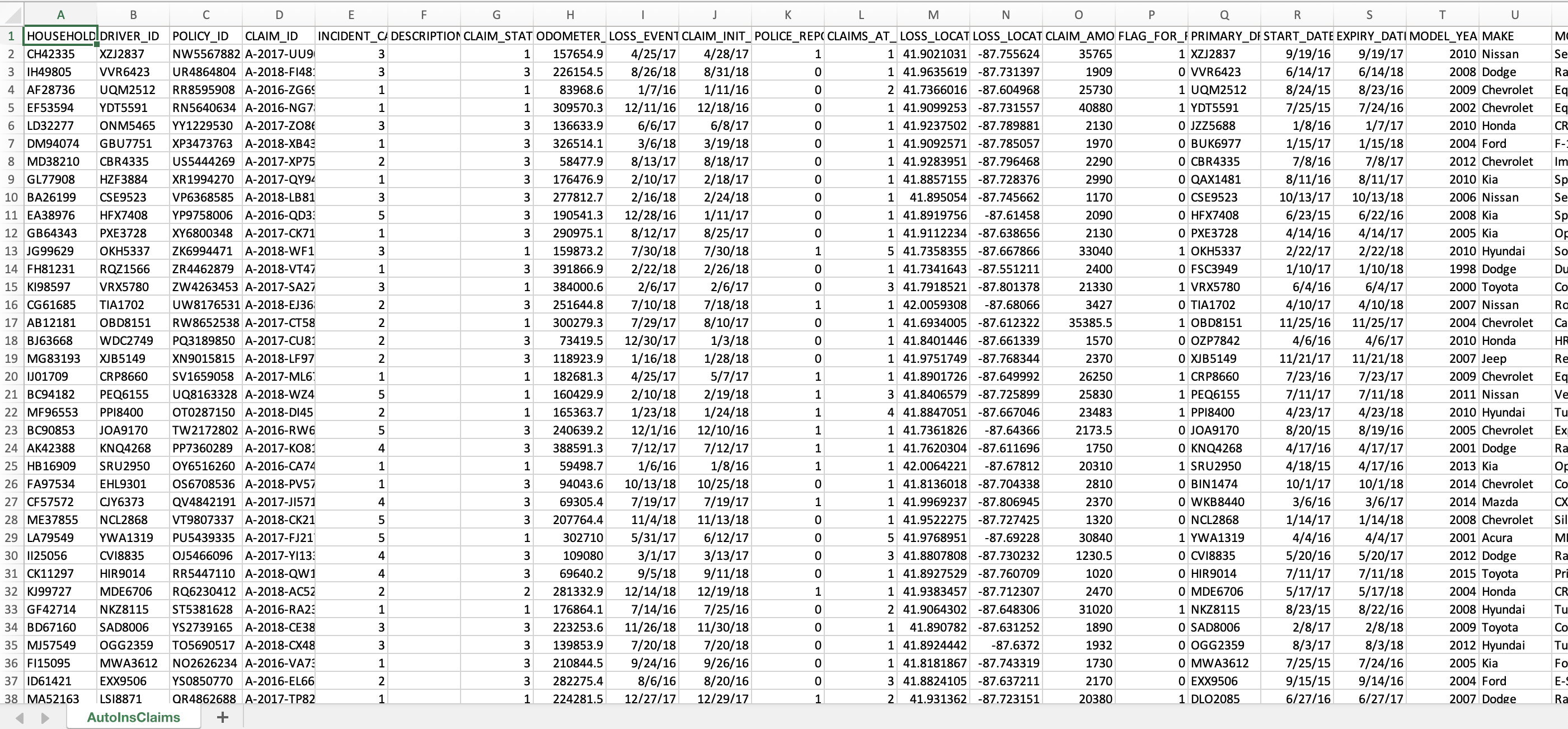
* IBM Watson Studio: Analyze data using RStudio, Jupyter, and Python in a configured, collaborative environment that includes IBM value-adds, such as managed Spark.
* IBM Auto AI: The AutoAI graphical tool in Watson Studio automatically analyzes your data and generates candidate model pipelines customized for your predictive modeling problem.
* IBM Cloud Object Storage: An IBM Cloud service that provides an unstructured cloud data store to build and deliver cost effective apps and services with high reliability and fast speed to market. This code pattern uses Cloud Object Storage.

## Understanding the Data

Let’s begin by accessing the CSV file required for the following steps.

1. Download the CSV file: <https://github.com/apischdo/Skillsbuild/blob/master/AutoInsClaims.csv>

It is recommended that you use Firefox, IE or Mozilla (Edge and Safari are not recommended). Once you access the Github page, click **Raw**. Then, from the ensuing page, right-click and save the file as CSV (change file type to **All file types**).



Open the file with Excel and notice that the columns are in string format and some clearly need to be numeric and that is just at a first glance. Let’s take a moment and understand the significance or more aptly, the predictability of each of the columns. Clearly, you can tell some of that info will not help you in predicting fraud. We need to remove the noise from the signal.

Take a moment, and note the column names: which may prove relevant? which are simply not needed? What if you used the date columns to calculate lapsed days from time of accident until reporting it? Or noted if the claims are being filled too close to the policy expiration date? Was there a police report? Let’s look at expired licenses at the time of submitting the claim. What about clients with low mileage discounts (7500 per year) that do not have low mileage?

Take a moment and consider the table below bearing in mind the questions asked above, let’s discuss that in a group if feasible, because before long, you will be immersed in feature engineering activities.

As you may have noted, there is a column named: **FLAG\_FOR\_FRAUD\_INV**. Think of this column as training data for a supervised machine learning system. The rows with a value of 1 (True) have been verified and classified as fraud. That is known, not a prediction. We are going to build a model using this “training set” for data to predict future behavioral patterns that may then be flagged as potential fraud on a new CSV file never seen before by Watson Studio.

|  |  |
| --- | --- |
| Feature name | R for remove  P for predictable  and why do you think that? |
| HOUSEHOLD\_ID | R |
| DRIVER\_ID | R |
| POLICY\_ID | R |
| CLAIM\_ID | R |
| INCIDENT\_CAUSE  1 = driver error  2 = natural causes  3 = other driver error  4 = crime  5 = other causes | K |
| DESCRIPTION | R |
| CLAIM\_STATUS  1 = open  2 = approved  3 = paid  4 = flagged for fraud  5 = denied  6 = appeal | R |
| ODOMETER\_AT\_LOSS | K |
| LOSS\_EVENT\_TIME | K |
| CLAIM\_INIT\_TIME | K |
| POLICE\_REPORT  1 = there was police report  0 = no police report | K |
| CLAIMS\_AT\_LOSS\_DATE (# of claims per individual) | K |
| LOSS\_LOCATION\_LAT | K |
| LOSS\_LOCATION\_LONG | K |
| CLAIM\_AMOUNT | K |
| FLAG\_FOR\_FRAUD\_INV | K (THIS IS YOUR X-AXIS) |
| PRIMARY\_DRIVER\_ID | R |
| START\_DATE | K |
| EXPIRY\_DATE | K |
| MODEL\_YEAR | R |
| MAKE | R |
| MODEL | R |
| PLATE | R |
| COLOR | YOUR CHOICE |
| INITIAL\_ODOMETER | K |
| LOW\_MILEAGE\_USE | K |
| FIRST\_NAME | R |
| LAST\_NAME | R |
| GENDER | K |
| BIRTHDATE | K |
| SSN | R |
| DRIVERS\_LICENSE\_ID | R |
| DRIVERS\_LICENSE\_EXPIRY | K |
| DRIVERS\_LICENSE\_STATE | WHAT DO YOU THINK? |
| DATE\_AT\_CURRENT\_ADDRESS | K |
| CONTACT\_NUMBER | R |
| EMAIL | R |
| COMMUTE\_DISCOUNT | K |

You are now ready for the detailed steps.

# Build an AI model using AutoAI

## Create a new Watson Studio project

1. Open an already provisioned Watson Studio service from prior labs
2. Click **Launch in IBM Cloud Pak for Data**.
3. In the ensuing pop-up dialog, click **New Project** and click **Next**.
4. Define the project by giving a Name. Since you have already created the Object Storage, then it should appear under the Storage heading.

Graphical user interface, application

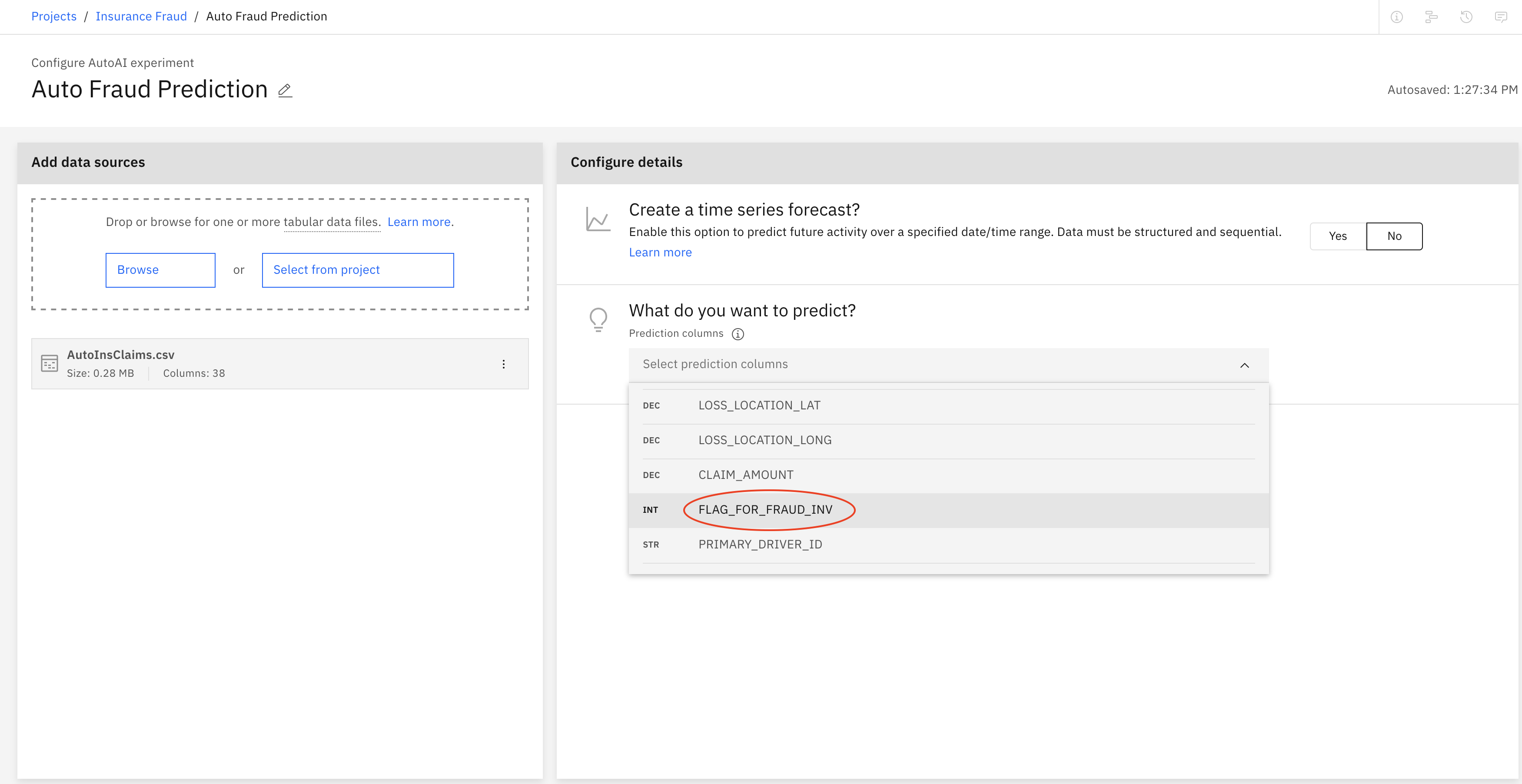
Description automatically generated

1. Click the **Assets** tab.
2. Click **New Assets** (blue box).

Graphical user interface, text, application, email

Description automatically generated

1. Select the **AutoAI** tile.
2. Specify a name for the experiment.
3. Click **Associate a Machine Learning service instance** to this project and select the Machine Learning service instance and click reload. If you do not have Machine Learning service instance, then follow the steps on your screen to provision the service.
4. Click **Create**.
5. Use the **Browse** button to upload your CSV file from your local drive.
6. Select **No**, since this is not a time series event.
7. Select **FLAG\_FOR\_FRAUD\_INV** as the predictable column.
   1. Try out other features (column headings) and observe how the system recognizes Regression and Multiclassification as other potential approaches.
   2. Before you click **Run Experiment**, discuss with your team and instructor the nuances of what lurks in the Experiment Settings tab.



At this point you need to perform certain experiment settings. Before you begin in earnest, it is worth while to see some of the magic behind AutoAI.

1. Instead of FLAG\_FOR\_FRAUD\_INV, select CLAIM\_AMOUNT. Notice that the Prediction type is Regression. Now, select a column that you think would reveal a multiclassification. You are on your own here. But do not save settings nor run experiment yet; you are taking various actions to see how the system responds.
2. Once you are done with your observations on few of these columns (features), go back and select FLAG\_FOR\_FRAUD\_INV
3. Before you run the experiment, click **Experiment settings**.
4. Include **Gradient Boosting Classifier** yet another estimator to run your experiment.

Graphical user interface, application

Description automatically generated

1. Scroll further down and you will notice that by default the system will select, based on its own decision, given the myriad of algorithms to choose from, will select either, one, two, three or four estimators or algorithms. For sake of time and other credit unit hour savings, accept the default of two algorithms:

A screenshot of a computer

Description automatically generated

1. Notice that the system is set to perform a 3-fold training and test set approach. This means that out of a possible 975 rows (records of individuals) the following events will take place:
   1. First, 10% of all rows (~98) will be kept away from the system at first…randomly chosen 98 rows.
   2. Next, the system, now has roughly 900 rows where it will break it up into three chucks (randomly grouped).
   3. AutoAI will build patterns from 2 of those three chunks and test itself on the third partition available to it. It may pick the top 1/3 and bottom 1/3 and test itself on the middle 1/3.
   4. This training set will then include two the bottom 1/3 partitions and test itself on the top 1/3 and so forth with the top and bottom 1/3, while testing itself in the middle section. The machine has the answers (supervised training) in column P (FLAG\_FOR\_INV).
   5. Once patterns are built, the system discards all training data and applied those patterns (weights) onto the hidden 10% that was kept away as test set. The metrics that we obtain from data unseen by the machine, yet obvious to human, is the telltale story behind the validation of the metrics.
2. A screenshot of a computer

   Description automatically generated
3. Move the slider up and down the line and notice the dialog box that appears as you move it above 15% test set. Bring it back to the original settings of 10%.
4. Click the **Data source** tab and start unchecking the features that you deemed unpredictable from the table above (bear in mind there are 4 pages in this selection). For this experiment, uncheck the **Claim Status** as well as all other private information (PII).
   1. Ensure to keep ALL fields that have something to do with date. Birthdays (some folks drive with expired license) or Date\_at\_current\_location (some folks tend to cheat on their auto insurance who tend to move from one address to another in a short time period). Keep all date-related columns.
   2. You may decide on your own if you want to keep or remove certain attributes such as Color. Hard to argue the point that folks with Cherry red cars are up to more mischief but might be interesting to see if that may be the case.
   3. Obviously, name, phone number, social security and other data entry ID fields are not predictable features as to why certain individuals may do fraud with their auto insurance. Furthermore, according to GDPR and the California Consumer Privacy Act such data is simply unlawful to tamper with.

Graphical user interface, application, Teams

Description automatically generated A screenshot of a search column

Description automatically generated

A screenshot of a search column

Description automatically generated

A screenshot of a search column

Description automatically generated

1. Save the settings.
2. Run the Experiment.

## Analyze results

In this example, AutoAI experiment generated two pipelines. The duration of experiment depends completely on the size of the dataset. AutoAI selects the appropriate machine learning algorithm (in the fifth stage of the process under Model Selection) which is best suited for the dataset.

Each pipeline is run with different parameters, pipeline 2 is run on a sequence of HPO (hyper parameters optimization) & FE (feature engineering) whereas pipeline 4 includes HPO (hyper parameters optimization), FE (feature engineering) and a combination of both. All these are done on the fly! Isn't it amazing that we just have to sit and watch while AutoAI takes care of things for us and generates awesome machine learning models!! There's very minimal intervention required to get things going and in no time, we have the generated pipelines to choose from.

A screenshot of a computer

Description automatically generated

Note that the system ‘decides’ which algorithms to choose. You can always uncheck certain algorithms and force it to use fewer estimators and observe the accuracy from that machine rendered decision.

Pipeline 8, using Snap Random Forest Classifier is the winner of the two.

1. Click the highest-ranking pipeline (with the star) to see the evaluation metrics on the left side.

A screenshot of a graph

Description automatically generated

Notice that the Receivable Operating Curve, pegs the True Positive against the False positive. The Receiver Operating Characteristic curve, is a graphical representation used to assess the performance of a binary classification model. It measures the trade-off between the True Positive Rate (Sensitivity) and the False Positive Rate (1 - Specificity) at various thresholds for a classification problem.

In a binary classification scenario, the ROC curve plots the relationship between the True Positive Rate (TPR) on the y-axis and the False Positive Rate (FPR) on the x-axis as the classification threshold is varied. The TPR represents the proportion of actual positive cases correctly predicted as positive by the model, while the FPR represents the proportion of actual negative cases incorrectly predicted as positive by the model.

The ROC curve provides valuable insights into the model's ability to discriminate between the two classes across different threshold settings. A well-performing model will have a ROC curve that hugs the upper-left corner, indicating high TPR and low FPR across a range of threshold values. The area under the ROC curve (AUC-ROC) is a common metric used to quantify the overall performance of a classification model. A higher AUC-ROC value suggests a better discriminative ability and, in turn, a better model.

1. Click the **Confusion matrix** in the left panel.

A calculator and a calculator on a computer screen

Description automatically generated

1. The system predicted that from a total of 98 records, it predicted that 37 fraud incidents may take place, yet it missed on 4 counts. It did very good with predicting that 57 of those 98 rows will not result in fraudulent activities and it did not miss on any of those predictions. Overall, an accuracy of 96% is quite impressive.
2. Click the **Feature summary** link and it is obvious now that excessive claim amount is the number one cause of auto insurance fraud.

A screenshot of a computer

Description automatically generated

1. Keep this chart handy, for this is how you calculate the evaluation metrics stemming from the confusion matrix:

A diagram of mathematical equations

Description automatically generated with medium confidence

As you have noticed we left out how to calculate the accuracy of the system. That is for you to add to your notes.

1. The last undertaking here is to save this model either as a model (pmml file) or as Jupyter Notebooks, which also can be run and used as a model for future datasets. Remember, we are not after people, merely human traits (or features) that may tell the tale of fraudulent activities.
2. Click the blue **Save as** button in the top right and select Notebook.
3. Click **Create**.

A screenshot of a computer

Description automatically generated

1. Click View in Project

A screenshot of a computer

Description automatically generated

1. AutoAI, does a wonderful job in creating a meaningful Python code in Jupyter Notebooks. Take a moment and scroll thru the python code.
   1. Note, the first thing that you include in the first cell, is inclusion of relevant libraries and installs
   2. Second, is the connection credentials with the NoSQL database
   3. Third, is a depiction of the settings that you picked in AutoAI, the accuracy of 10%, the x-axis and the list of algorithms for the system to consider.

A screenshot of a computer

Description automatically generated

* 1. You can now save this a local file onto your local system. From the File menu, select Download as.. and then click the **ipynb** file type.

A screenshot of a computer

Description automatically generated

1. We encourage you to find your own dataset and upload that instead of this CSV file. Perhaps you can predict which factors may contribute to students not completing their program; or is there an uptick of auto accidents at certain hours or certain locations.