IBM Data Science Capstone

CRISP-DM Report  
Nathan Walker  
9-5-20

# Business Understanding

Imagine you are a 911 dispatcher. A call comes in, and it’s a frantic man on the other end of the line, saying there has been a car accident and they need help, but then the line drops. You know that your department has limited resources, and you need to triage the severity of the accident so that you know who to send to the scene of the accident. It wouldn’t be smart to send the entire fire, police and EMS services for what’s likely just a fender bender, but if it’s a major accident, seconds matter. What do you do?

What if machine learning could help you make that decision? That is the reality of our new data driven world.

As a former firefighter, I know first-hand how important it is to allocate appropriate resources to an emergency situation. Personnel constraints, equipment shortages, and budget constraints can all play a role in those decisions. If there is a major fire going on and a call comes in for a wreck, it’s not always practical, or even possible, to send all of your resources. Especially for just a small fender bender at noon on a Tuesday. But if there is an accident on a twisty backroad at 2AM on a rainy Saturday, you should expect the worst and send whoever you can.

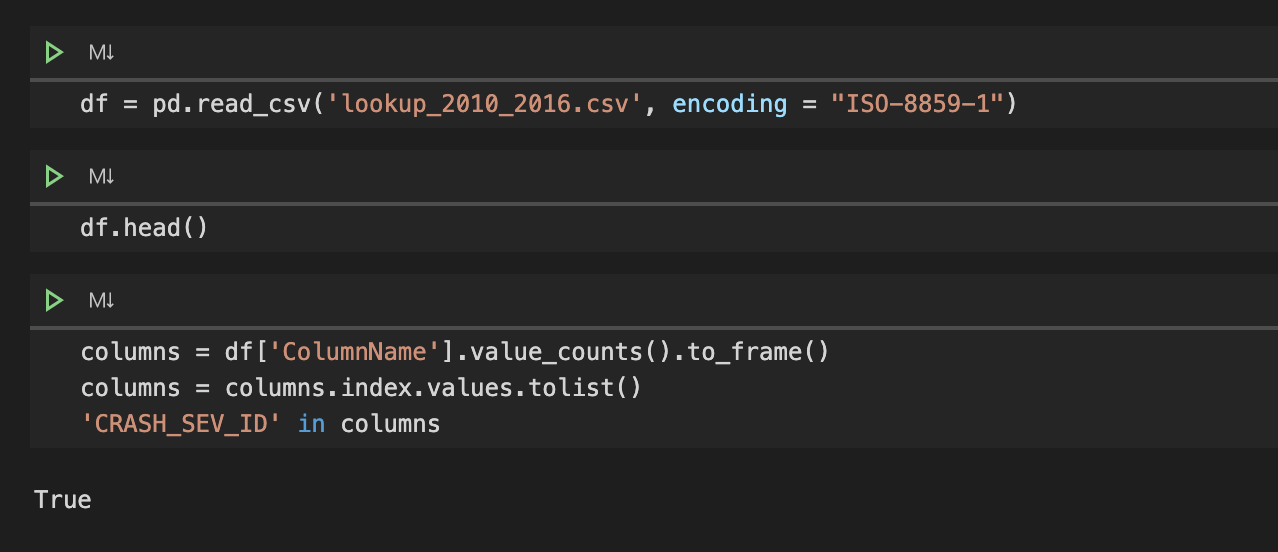
And that’s the goal of this project; to build a model to predict the severity of an accident, so that it can help triage accidents based on historic data.

# Data Understanding

I’m a resident of the great state of Texas, so I chose to build my model using data from my state. While researching for data I could use, I found that public crash records do indeed exist for my state.

<https://data.world/spatialaustin/texas-crash-records-information-system-cris-extract>

I downloaded the 'field lookup' file to see if "crash severity" was a field I could target.



Success! It looks like this data has what we need to build a model to predict the severity of a crash. And there are many good potential predictor columns in the data; for example, the names of the roads, the date and time, the road’s surface material, the weather conditions, and much, much more. We should certainly be able to build a good classification model based on this data.

Unfortunately, the data here was fairly stale, with records up to only 2016. The latest data can be accessed from the Texas Department of Transportation for statistical purposes. Accessing this data requires signing up for the CRIS database.

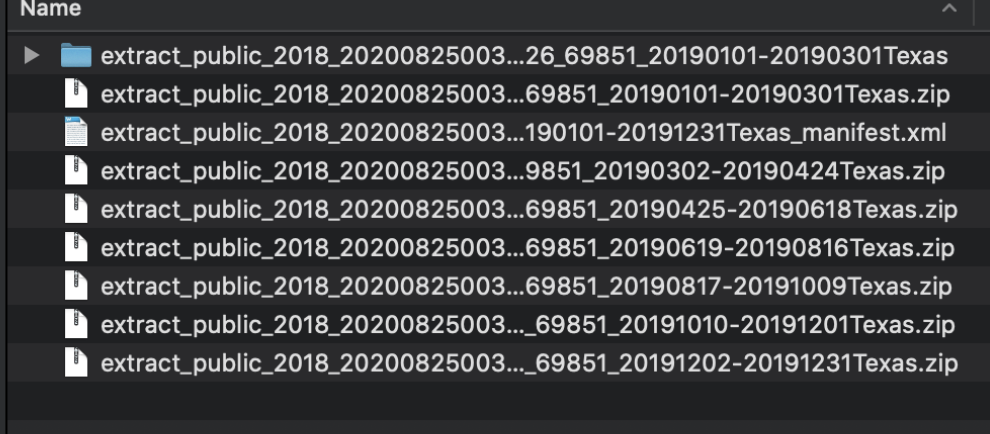
https://www.txdot.gov/government/enforcement/data-access.html

http://ftp.dot.state.tx.us/pub/txdot-info/trf/crash\_statistics/automated/cris-guide.pdf

https://cris.dot.state.tx.us/secure/Share/app/home/welcome

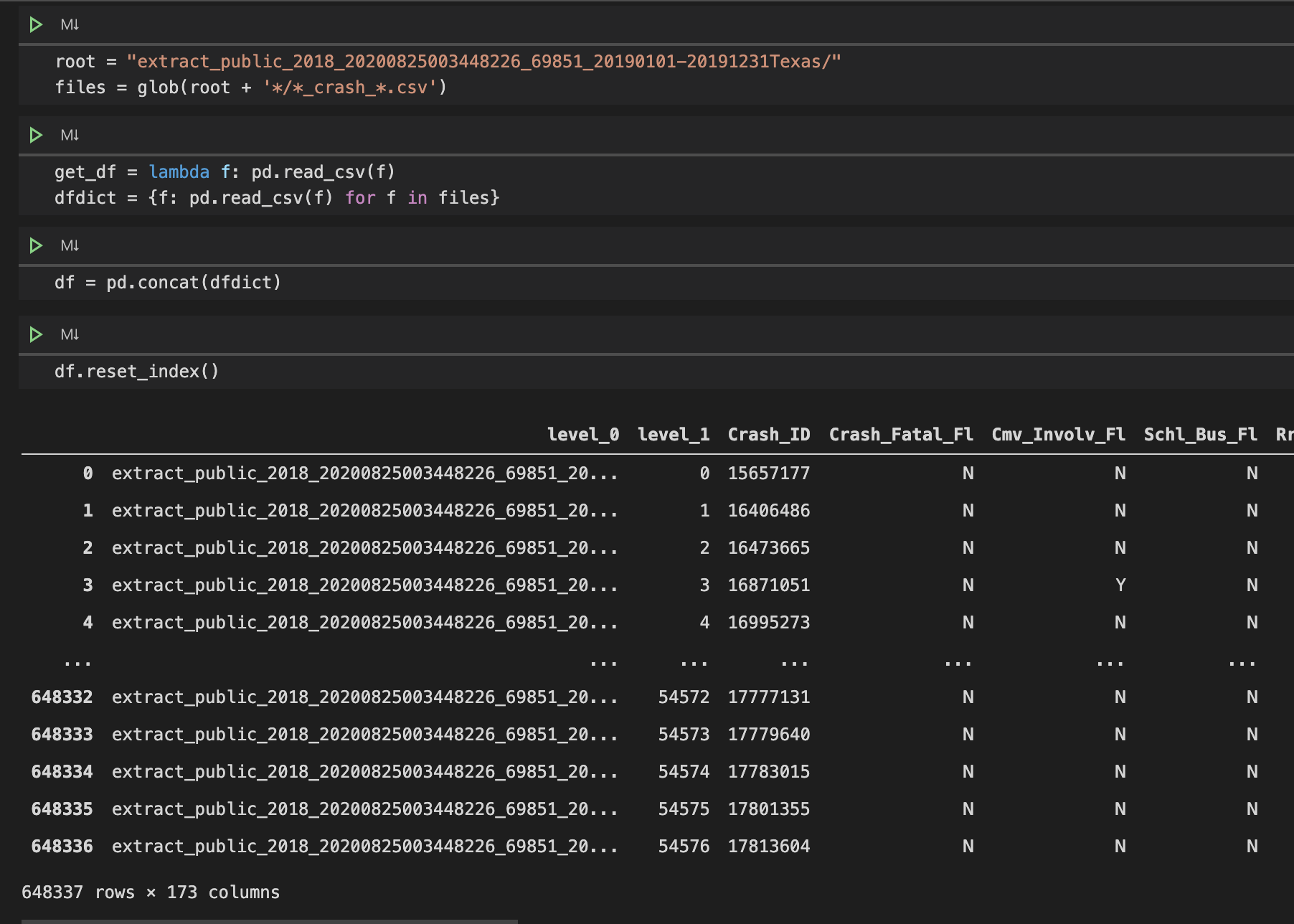
You can only request one year of data at a time. I will be using data from the year 2019 (1/1-12/31) to build my model.

It took about a day to get the data from the CRIS team, but once I did, it came in an encrypted zip file. Inside of that were several more zip files, and judging by the filenames, it's the data broken up into smaller date ranges.



I unzipped the first archive, and poked around the data. The file 'extract\_public\_2018\_20200825003448\_crash\_20190101-20190301Texas.csv' had the target data, CRASH\_SEV\_ID.

My first task was to extract all of the data and pull it into a pandas dataframe.



This took me a while to figure out! The glob function was something I had not personally used before, but it allowed me to quickly import all of the relevant files all at once.

I now have a dataframe with 648337 and 171 columns. Let’s discuss the columns.

An extremely useful file for understanding all of the data is available here:

<https://www.txdot.gov/government/enforcement/data-access.html>

<http://ftp.dot.state.tx.us/pub/txdot-info/trf/crash_statistics/automated/publicextractfilespecification.xlsx>

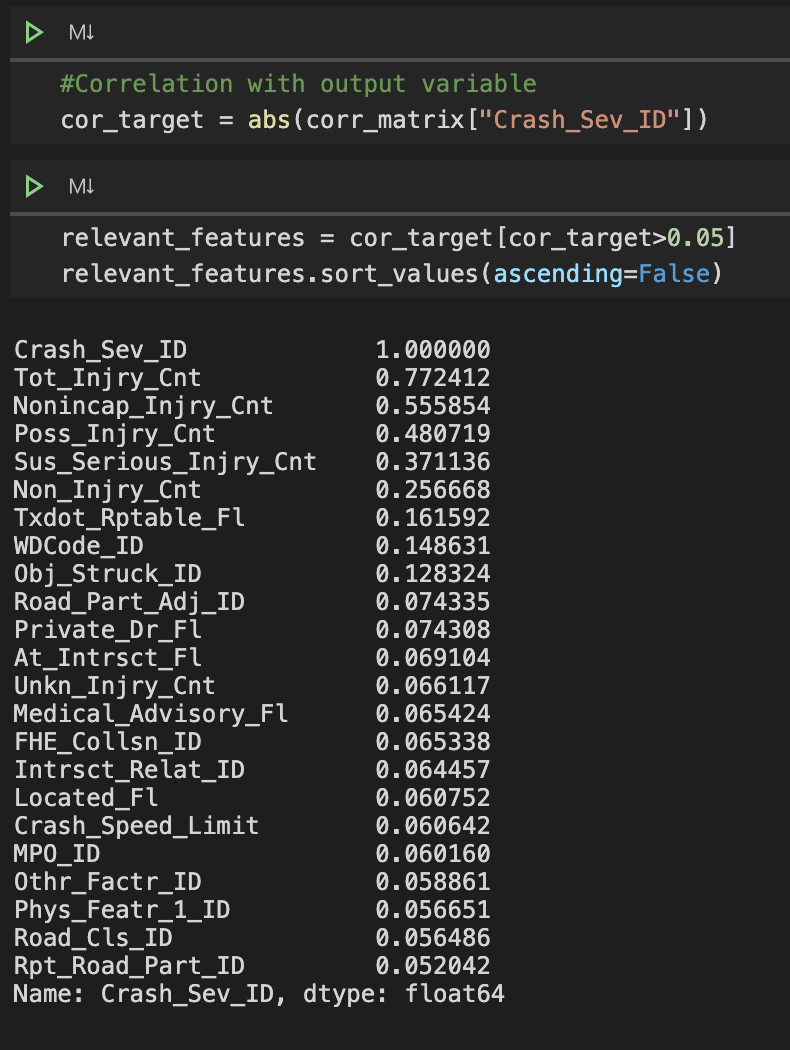
This file has detailed descriptions of all of the columns, and what each value represents.

The vast, vast majority of this data is going to be irrelevant to our problem. We need to predict the severity of a crash, and the factors that contribute to the crash are all we are concerned with. We need to dig through the list and make sure we understand each column.

I suspect that what we need to do is break this list into two columns - items that were conditions leading up to the crash, and items that are effects of the crash. For example, road conditions are likely contributors to the crash, while the damage to the vehicle is a result of the crash. Let's call these ‘contributors’ and ‘results’.

There are also going to be completely irrelevant factors, like the name of the investigating service, that we can remove entirely.

First, I wanted to find what columns were the mostly highly correlated with the crash severity. This proved to be less interesting than I thought it would. The most highly correlated items were what I would consider the ‘results’ of a crash, and not really a ‘contributor’.



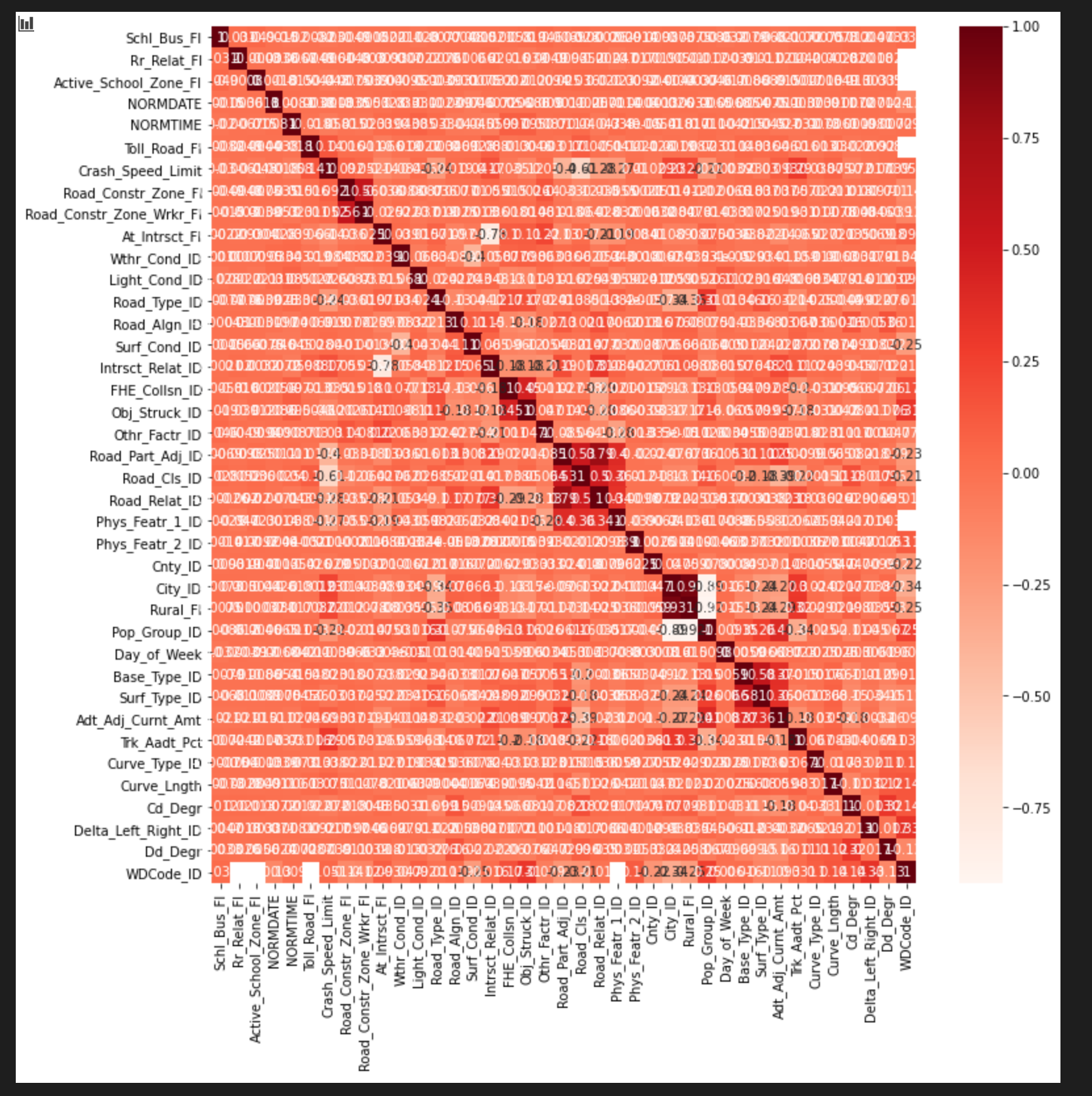
The most highly correlated fields were all of the injury count fields – this is not information you will have necessarily until you get resources to the crash scene. You can’t predict how bad a crash is GOING to be with data that isn’t generated until AFTER the crash. Unfortunately, you are typically looking for fields that have at least some correlation, but we aren’t going to have much luck with this data. All of the contributor fields have an absolute value of less than 0.15 correlation.

Since most of the fields are not highly correlated with the crash severity directly, tells me that we won’t be able to use a linear regression model or similar; we should use Decision Tree, SVG, or Random Forrest for our model. Which make sense, since this is a categorization problem.

After working through all of the fields, and doing lots of manual analysis of the 171 columns, I found the following columns to be most relevant, at least in my mind.

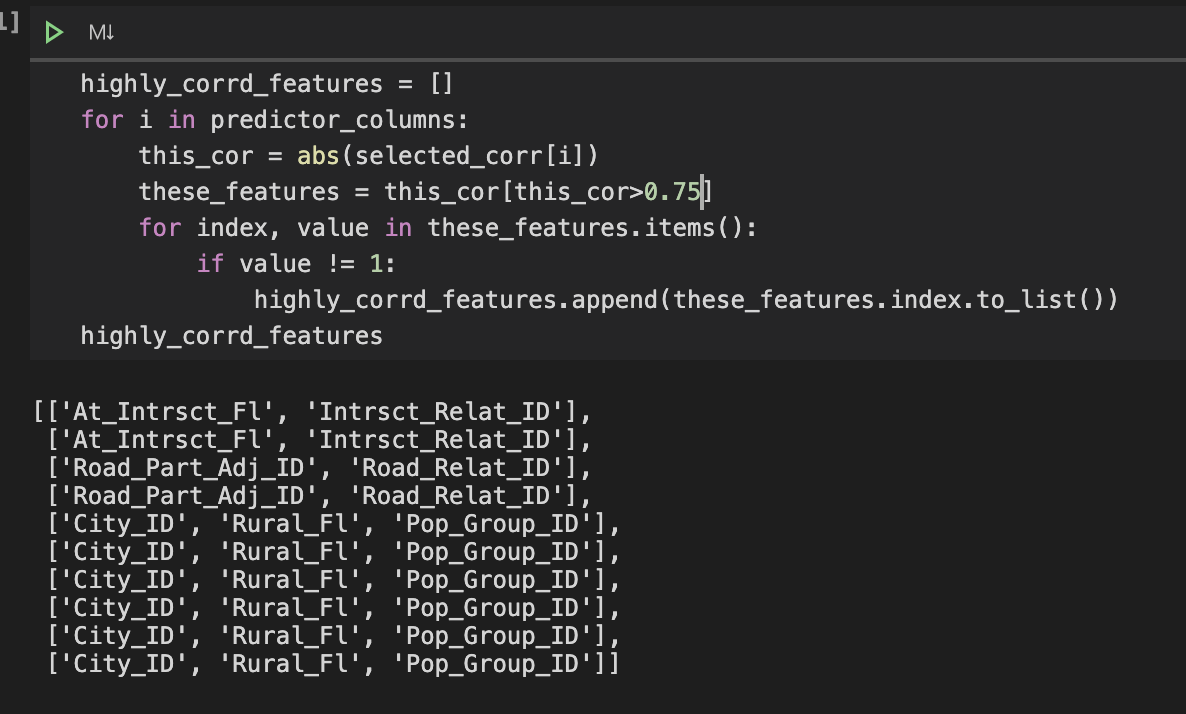
|  |  |
| --- | --- |
| **Toll\_Road\_Fl** | **Toll Road/Toll Lane** |
| **Crash\_Speed\_Limit** | **Speed Limit** |
| **Road\_Constr\_Zone\_Fl** | **Construction Zone - Indicates whether the crash occurred in or was related to a construction, maintenance, or utility work zone, regardless of whether or not workers were actually present at the time of the crash** |
| **Road\_Constr\_Zone\_Wrkr\_Fl** | **Workers Present - Indicates whether workers were present in the road construction zone at the time of the crash** |
| **At\_Intrsct\_Fl** | **At Intersection - Indicates if the crash occurred at an intersection.** |
| **Wthr\_Cond\_ID** | **Weather Condition - The prevailing atmospheric condition reported by the officer at the time of the crash** |
| **Light\_Cond\_ID** | **Light Condition - The type and level of light that existed at the time of the crash** |
| **Road\_Type\_ID** | **Roadway Type** |
| **Surf\_Cond\_ID** | **Surface Condition - The surface condition (wet, dry, etc) present at the time and place of the crash** |
| **Intrsct\_Relat\_ID** | **IF- Intersection Related - Specifies whether a crash occurred at an intersection, not at an intersection, or if the presence of an intersection contributed to the crash** |
| **FHE\_Collsn\_ID** | **IF- Manner of Collision - The manner in which the vehicle(s) were moving prior to the first harmful event.** |
| **Obj\_Struck\_Id** | **IF- Object Struck - Object Struck is an obstruction in, on, or around a road that a motor vehicle involved in a crash has made contact with.** |
| **Othr\_Factr\_ID** | **IF- Other Factor - Additional detail of events/circumstances concerning the crash** |
| **Road\_Part\_Adj\_ID** | **IF- Roadway Part - The part of the roadway on which the vehicle(s) was traveling prior to the crash.** |
| **Road\_Cls\_ID** | **IF- Road Class - The functional classification group of the priority road the motor vehicle(s) was traveling on before the First Harmful Event (FHE) occurred** |
| **Road\_Relat\_ID** | **IF- Roadway Relation - Roadway Relation refers to where the First Harmful Event (point of impact) occurred in relation to the roadway.** |
| **Phys\_Featr\_1\_ID** | **IF- Physical Features - Physical Features fields 1 and 2 describe roadway features which were a factor in the crash** |
| **Phys\_Featr\_2\_ID** | **IF- Physical Features - Physical Features fields 1 and 2 describe roadway features which were a factor in the crash** |
| **Surf\_Type\_ID** | **Surf\_Type\_ID - The road surface type (concrete, blacktop, etc) at the location of the crash, for crashes located on the state highway system** |

However, I wanted to make sure I didn’t have any redundant fields. To further reduce my dimensionality, I did a correlation heat map of the data.

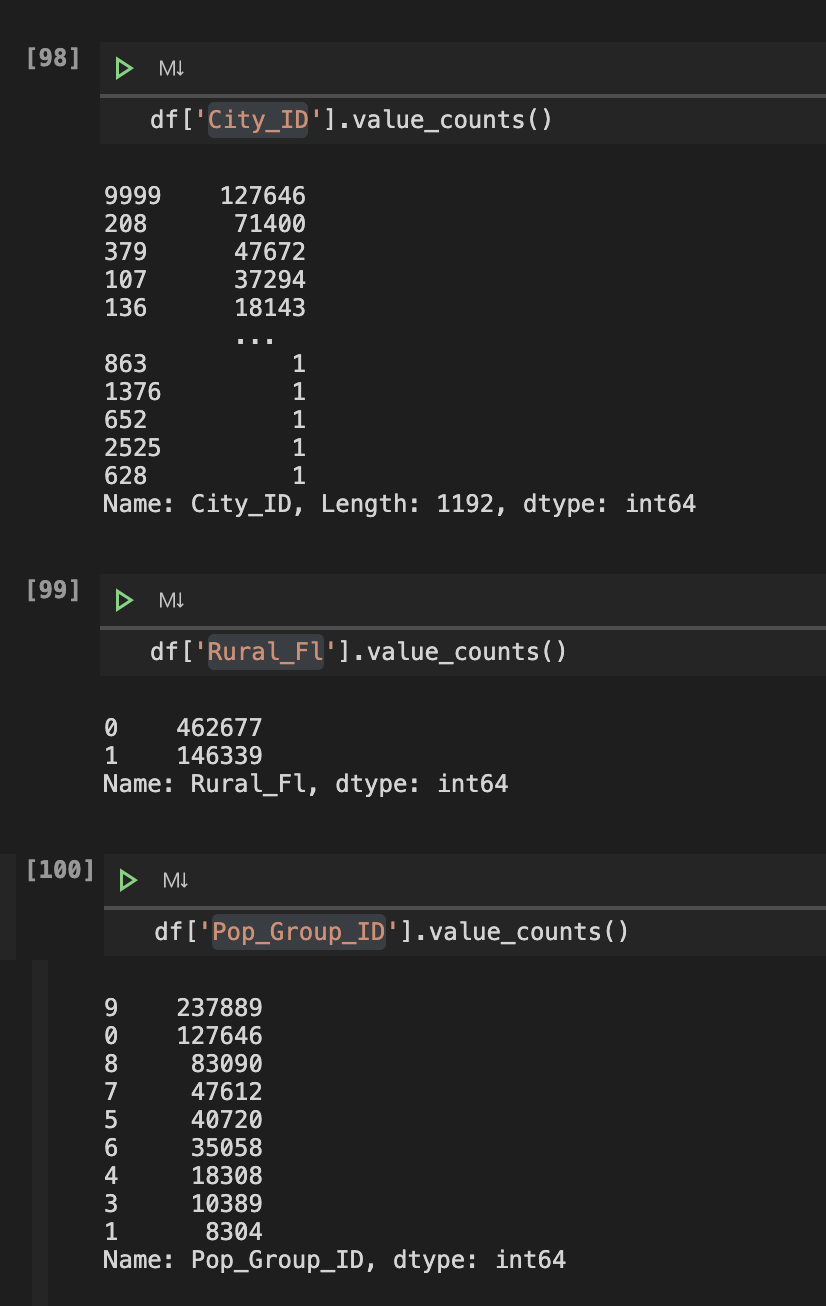


There are indeed several fields that are highly correlated.

I ran a quick loop to pull out exactly which fields were the most highly correlated.



Pretty interesting. The fields City\_ID, Rural\_Fl, and Pop\_Group\_ID are all over 0.8. Let’s look more closely at these fields.



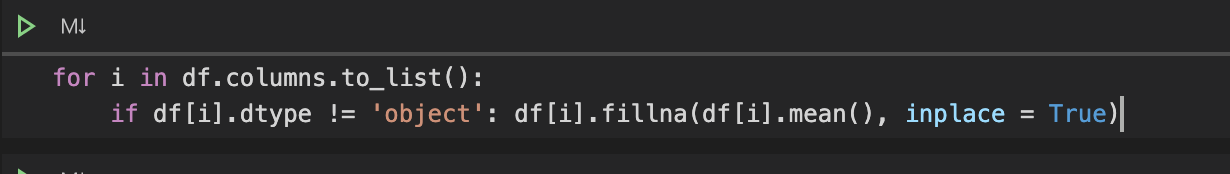
We can see that Rural\_Fl may not be that useful. It’s really only a True/False field that does not narrow down our data very much. City ID is very interesting, but there is so much variability in the data (and a lot of 9 – Not reported), that it may yield inaccurate results.

Pop\_Group\_ID however, could be considered like a version of “binning” the City\_ID. It should provide me with enough granular detail to make a good prediction, but generalized enough to not over fit. We’ll drop the other two columns.

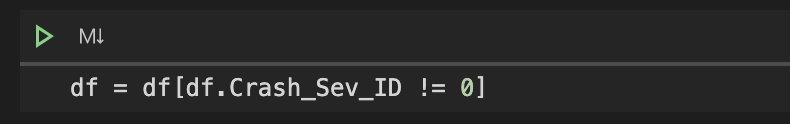
From here, much of the data is going to need to be normalized – for example, Y and N values will be replaced with 1 and 0 respectively, and of course values like dates and times will need to be converted to a float.



We need to fill any NaN values as well. The technique we’ll use for now is inserting the mean value. We’ll revisit this during our modeling and see if any other techniques are more effective, but for now this will suffice.



There are also records where the crash severity is 0, which means there was no data for that incident. We’ll just drop these records entirely.

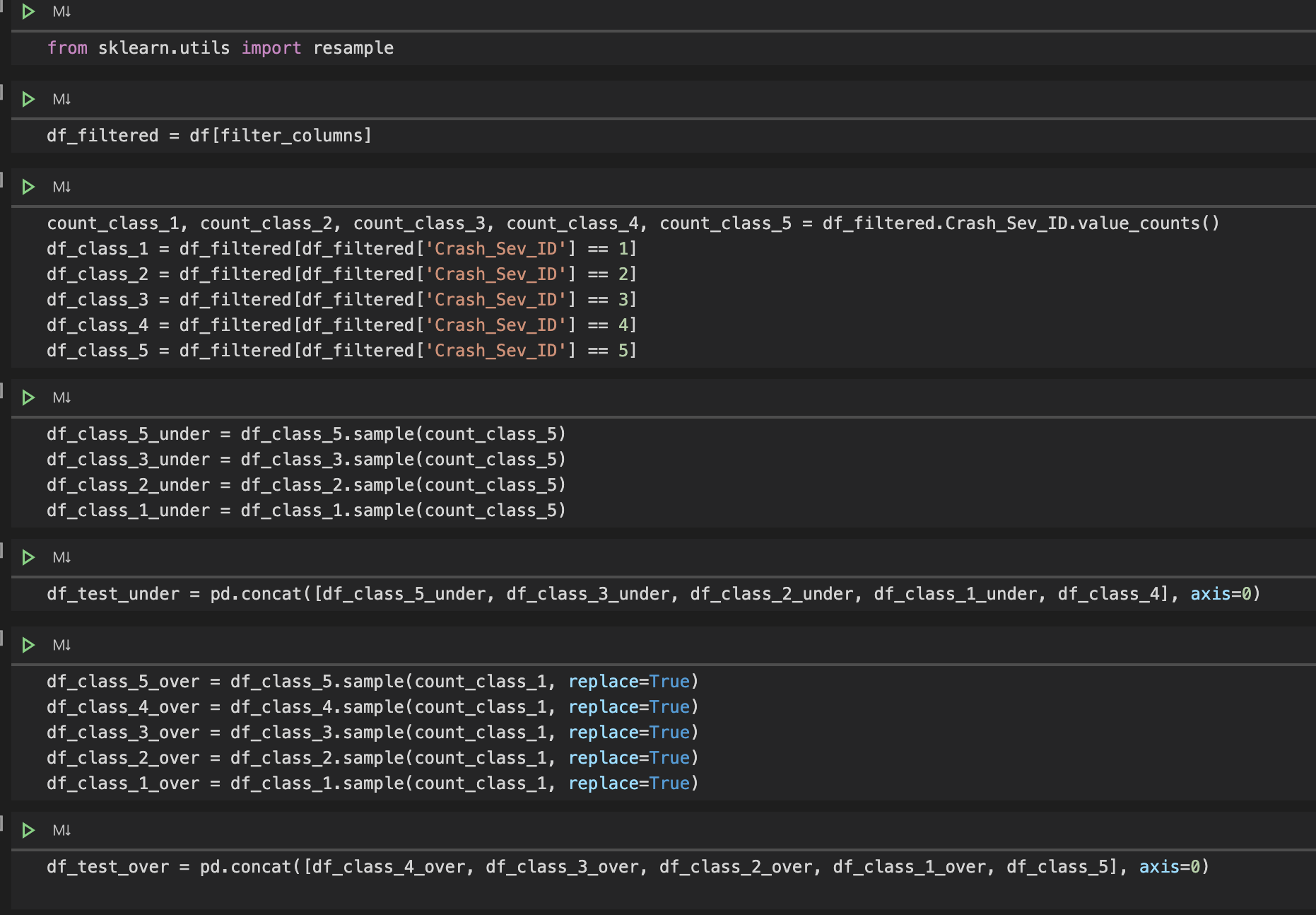


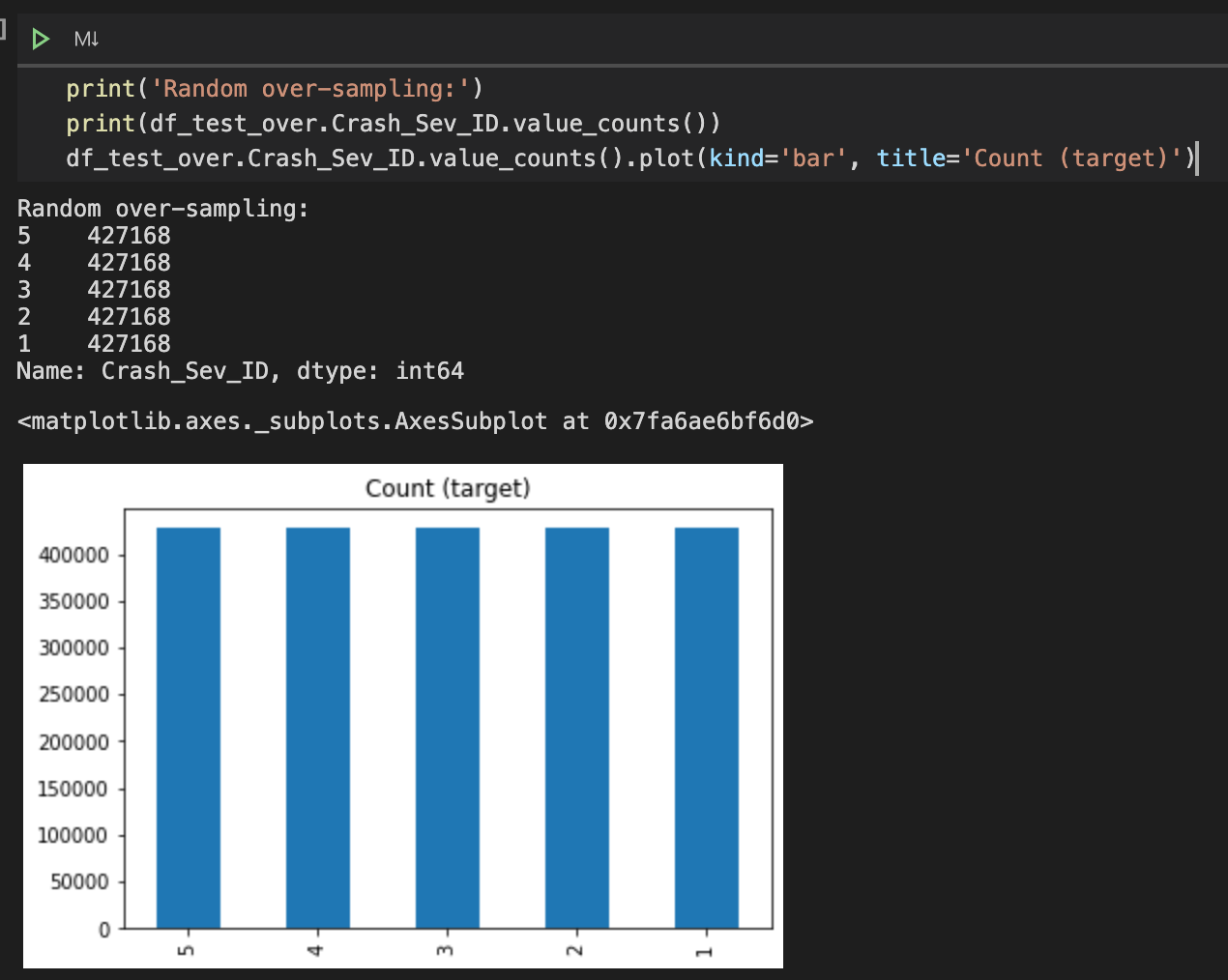
The next major factor to take into consideration is, how balanced is the data?

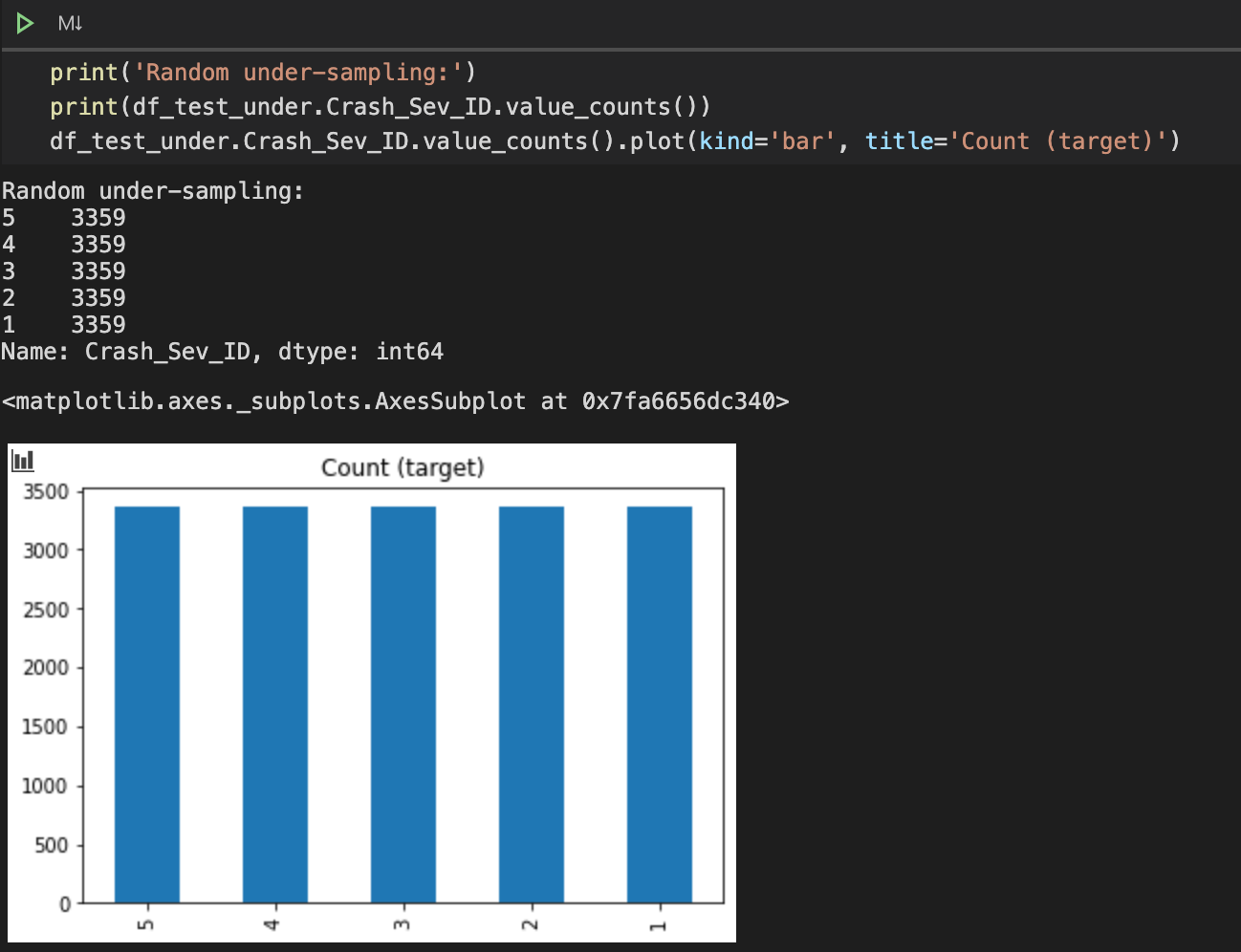


This data is clearly skewed towards the value 5. This is going to cause overfitting.

We’ll use sklearn’s resample library to create two new data frames, one with over-sampling and one with under-sampling. The oversampling frame will take our under-represented values and randomly up-sample our values to match the highest value, and the under-sample frame will randomly remove rows from the over-represented values. Both of these techniques have their shortfalls; potentially overfitting and loss of information, respectively. But it will give us much more accurate results than not balancing it at all. We’ll train models with both and evaluate them.







We should be ready for modeling.