**Introduction/Business Problem**

The area I’ll investigate is about accidents and traffic flow at the highways. A lot of incidents can happen and impact whether the traffic goes smooth or not. Severe car crashes are one of them.

Based on a set of conditions like weather, light, road conditions and more, we’ll build and train a model to forecast emergency situations. This is to alert drivers for high risk so that they will adjust to drive more carefully or make other choices – like taking the train or staying home.

So, the idea is to predict the likelihood of accidents in regard to the traffic flow. As I’ve chosen to work with the shared dataset, I’ll be analyzing data from the Seattle Department of Transportation. The model should make sense in most other places too, but it’s probably important to be aware of the need for local adaption. What independent variables that should be used in the local model could vary.

We’ll build our model based on Machine Learning to predict the risk for severe accidents. Supported by regression analysis we’ll clean the data to make sure we have the best possible set of variables to work with. Along the way we must make other tests and evaluations to choose the best possible tools/models to work with in our approach to optimize our results.

Our project should be of great value to both drivers and traffic-related governments and organizations, in perspective of the push for safer streets. This expanded to Seattle's "Vision Zero" in 2015, after Vision Zero has proved successful across Europe - originally implemented in Sweden in the 1990’s. Many cities across the U.S. have signed on to the idea that even just one death caused by these accidents is unacceptable and preventable. This study will hopefully reveal what, if any, measures we can take as individuals and municipalities to make travel in Seattle safer.

**Data**

I chose to work in Anaconda and had to read in the shared data file with ‘data = pd.read.csv(‘filename’)’. That gave me some problems as the error “module ‘pandas’ has no attribute ‘read’” was returned. Research pointed at the pandas.py file could make some confusion, but I was not able to fix it.

Workaround was to turn to IBM and their environment, where I could drop in the data file. I still encountered some problems with reading in the data, probably due to some displacement of the columns/cells/delimiters when saving the file locally. This time I was able to fix it by adding some code in the read function:

# df = pd.read\_csv(body)

df = pd.read\_csv(body, header=None, sep='\n')

df = df[0].str.split(',', expand=True)

The imported data then had the shape (194674, 4), so I needed to extract each of the 4 columns and then concatenate so that I had all data in one dataframe (194674, 82). Now it’s quiet obvious that there is several columns I don’t need to keep working with, but I need a better understanding of what data is in the respective columns to start cleaning. I’ll investigate them at a high level and make a rough selection:

* Selected: SEVERITYCODE, X (LONGITUDE), Y(LATITUDE), ADDRTYPE, SEVERITYCODE, COLLISIONTYPE, PERSONCOUNT, PEDCOUNT, PEDCYLCOUNT, VEHCOUNT, INCDATE, JUNCTIONTYPE, SDOT\_COLDESC, INATTENTIONIND, UNDERINFL, WEATHER, ROADCOND, LIGHTCOND, SPEEDING, ST\_COLCODE, ST\_COLDESC
* Not selected: OBJECTID, INCKEY, COLDETKEY, REPORTNO, STATUS, INTKEY, LOCATION, EXCEPTRSNCODE, EXCEPTRSNDESC, INCDTTM, SDOT\_COLCODE, PEDROWNOTGRNT, SDOTCOLNUM, SEGLANEKEY, CROSSWALKKEY, SEVERITYDESC, HITPARKEDCAR and all columns in the original column 2, 3 and 4

At this point I’ve two questions up for discussion:

1. I had this conception of working with regression and correlation analysis to identify the set of variables that influenced the probability for severe car accidents most. But I’ve realized that was a sort of misconception. On one hand since all variables needs to be numeric to work with correlation, the data would need to be prepared quiet a lot to be converted to numeric values (that could probably be done with the .get-dummies function). But also, on the other hand it seems obvious that there’s 3 columns with conditions that will make up the featured data, so the correlation idea is not relevant.
2. When selecting variables to include in the featured data set, I not only chose all the condition variables, but also several variables I had an idea I could use to predict the size or type of accident – like number of cars involved or where in the traffic the accident would be most likely to happen. I’m pretty sure that this would be possible/useful, but I’m delighted to deem it out of scope. Which means I should split my selected data into a) conditions that appears before the accident and b) characteristics of the accident and work further with a only.
   1. SEVERITYCODE, WEATHER, ROADCOND, LIGHTCOND
   2. X (LONGITUDE), Y(LATITUDE), ADDRTYPE, COLLISIONTYPE, PERSONCOUNT, PEDCOUNT, PEDCYLCOUNT, VEHCOUNT, INCDATE, JUNCTIONTYPE, SDOT\_COLDESC, INATTENTIONIND, UNDERINFL, SPEEDING, ST\_COLCODE, ST\_COLDESC

And then, what about missing data? There is no missing data in the severity code column, but in the 3 others it’s a lot. I think I should remove rows with missing values. Looking further into this I get that not only NaN occurs but also categories as ‘unknown’, ‘other’ and ‘ ‘, which is not useful as a condition, so I probably remove rows with these values as well. Yes, doing so gives me a dataframe with the shape of (23985, 4) which will the basis for further work.