ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования

Национальный исследовательский университет
«Высшая школа экономики»

Факультет экономических наук
Образовательная программа «Экономика»

КУРСОВАЯ РАБОТА

«Байесовское моделирование и Несбалансированная Классификация Для Предсказания Тяжести Автомобильных аварий»
«Bayesian Modeling And Imbalanced Classification For Predicting Severity Of Car Accidents»

Студент группы БЭК181
Щекотов Иван, Тамогашев Кирилл

Научный руководитель:
Борис Демешев

Москва 2021

Contents

1 Introduction

"Vision Zero" is a set of innovative road safety policies, aimed at reducing traffic accidents, which result in severe injuries and deaths. This strategy had been proposed by Swedish Road Administration in 1995 [1], has been employed since then in Decade of Action for Road Safety declared by United Nations in 2010 [1] and is believed to be the reason of why Scandinavian countries maintain one of the lowest death and injury rates in car accidents in the whole world[1]. "Vision zero" suggests relying on vast amount of data to take effective measures towards the goal, e.g. select dangerous junctions and lanes, build safer roads, decrease speed limits, exclude human error, separate roads from pedestrians and cyclists. Each country employs its own policy with regards to "Vision Zero".

In Russia "Vision Zero" is being implemented in a form of the national project[2], however the results are quite unsatisfactory yet: despite number of severe injuries and deaths decrease in Moscow (which in fact does not develop any strategies according to the national project), all other regions do not exhibit positive results. In our paper we try to explore factors that contribute to mortality rates in capital region.

2 Problem Statement

Car accidents may occur due to many reasons. And usually there are multiple factors determining the outcome of a particular crash. Among these factors are time of a day, weather conditions, car model, etc. The outcome of a crash, either tragic one or without any loss of lives and severe injuries, is supposed to correlate with the type of violation that caused the accident. The rationale underneath is straightforward: as it happens, the accident occurs after someone violates the law and causes that accident. If the violation is minor, for example, parking in the wrong place, it hardly can cause any significant injury, however other violations, such as crossing the solid line and driving onto the opposite strip may be the cause of significant injuries and deaths. So, the main question of this paper is to find out the extent to which factors like whether conditions, road conditions, type of violations do correlate with the severity of car accident and can be used as predictors to determine the outcome of a crash.

To answer the stated question, we examine the data on car accidents and car crashes in Moscow, collected from 2015 to 2021. The dataset contains information on more than 55000 accidents with all the required information. We carefully preprocess the dataset to group the features and the whole procedure of data preprocessing is described in chapter 3.

First and foremost, it is important to define a target variable. And as far as the aim of the research is to determine the influence of the factors mentioned above on the outcome of the crash, we should carefully pick the variable we will attempt to predict. There are three potential candidates: «deaths», «injuries», «severity». In the end «Severity» was chosen as a target variable and there are two reasons for that. Firstly, deaths are not a good target, as we cannot properly compare many injuries and one death without referring to severity. Moreover, we cannot even compare no deaths and one injury as well. Because one injury may be minor or hard, some need to refer to severity of the accident as well. Secondly, following the same logic we cannot properly use injuries as an indication of the significance of the accident, because it is impossible to directly infer the significancy of wounds. Based on that logic «severity» was picked as a target variable, and it was

[1] Data is given for 2015 in the following table.

[2] Национальный проект "Безопасные и качественные дороги"

preprocessed assigning «0» to light wounds, «1» to major wounds and deaths. It is worth noting that we do not discriminate between major wounds and deaths because the data does not allow us to properly distinguish two cases. Difference may exist because the ambulance was a couple minutes faster in one case or due to other factors omitted in the initial data. And as two categories are quite close to each other in terms of given features we are going to treat them as one case.

In the end the problem is defined as classification of cases into two groups. However, before diving into classification problem we perform the analysis of the target variable. If you look at the variable "severity" plotted monthly, you will see a clear downward trend. Nevertheless, it is clear that «severity» is an aggregate variable for «deaths» and «injured». So, we look at the behaviour of the two latter indicators. As it turns the rationale for the number of severe accidents going down is the decline in death rate. So, me construct several models to analyze that trend and emulate the behaviour. The detailed description of that model is shown in the fifth chapter.

Afterwards, we turn to classification problem. As it will be shown later in chapter four, we face a serious problem with imbalance in data accompanied by a significant overlap of two classes. In order to obtain meaningful result, we perform several models. First, we use simple logistic regression and `CatBoostClassifier` with class weights. Second, we implement three models specifically designed to deal with imbalanced classes: One-Class SVM, Isolation Forest and Local Outlier Factor. The detailed description of the outcome is presented in chapter five.

3 Data Preprocessing and Description

For our purpose we have chosen data gathered from January 2015 to April 2021 in Moscow. Data is represented via `geojson` format and contains extensive amount of features describing various aspects of car accidents, including coordinates, lighting, weather and road conditions, nearby objects, datetime, severity, information about vehicles, drivers and passengers involved, their health conditions in the aftermath, rule violations that caused accident, injured and dead counts[3]. Almost all of the described features fall into the categorical type and are of nested structure (i.e. there are multiple cars, people, road conditions in an accident) and they are non-hierarchical, which means simple label encoding is not applicable.

3.1 Generalizing and encoding unique values

The problem with data is that due to it's nature, some of the features have an enormous amount of unique values which are not relevant in all their variety. Our methodology insists on merging these unique values for every such category into groups by some inherent property that these values possess, e.g. if a person dies after being hospitalized, we prefer not to distinguish between various time spans over which death has occured or if there are various types of passenger cars, there is no reason to analyze them as being different. In another words we try to explicitly project data into lower dimension, because it is impossible to work with otherwise. We discuss these features further and describe ideas which led us to unification. All code that refers to generalizing unique values is presented in `utils` folder[4].

[3] Example of how data is structured can be found here.

[4] https://github.com/isdevnull/cw3/tree/dev/utils

3.1.1 Violations[5]

There are 104 unique violations that were ascribed to accidents. Violations are commited by drivers or by pedestrians (people that are not driving behind the wheel of the vehicle and that are not passengers) and they can be divided into 8 groups:

1. violations of driving with respect to car motion
2. violations of goods transportation or carriage of passengers
3. violations of obligations or non-compliance when driving a motorcycle
4. improper use of light signals to control traffic
5. violations commited by pedestrians
6. non-compliance with rules of safety when driving
7. violations of vehicle operation
8. other

There is no unified classification provided by Russian department of transport, so our division may be incorrect. It is based on sane reasoning and similarities between different events.

3.1.2 Nearby Objects[6]

Nearby objects represent buildings surrounding an accident and road type (i.e. junctions, pedestrian crossings, etc.). There are 58 unique values and they can be divided into 8 groups:

1. other
2. unmarked and marked junctions
3. unmarked and marked pedestrian crossings
4. places with increased transport density (usually, some kind of stop on road)
5. crowded places (e.g. bus stop)
6. controlled junctions
7. controlled pedestrian crossings

'Other' mostly contains various types of buildings, while other groups relate to some road objects. We didn't break 'other' into multiple groups because considered this step irrelevant for our purposes or reckoned that groups are rather small to be represented (both by accident occurence and unique values).

3.1.3 Automobile Categories[7]

Different types of vehicles (82) were divided into the following groups:

1. passenger cars
2. elite passenger cars
3. trucks
4. public transport
5. offroad and heavy-duty vehicles
6. motor vehicles with less than 4 wheels
7. other

3.1.4 Health Status[8]

The consequences of accidents are characterized by different health status (41). We define the following groups both for drivers, passengers and pedestrians:

[5] https://github.com/isdevnull/cw3/blob/dev/utils/violations.py
[6] https://github.com/isdevnull/cw3/blob/dev/utils/nearby.py
[7] https://github.com/isdevnull/cw3/blob/dev/utils/transport.py
[8] https://github.com/isdevnull/cw3/blob/dev/utils/health_status.py

1. no injuries
2. light injuries
3. wounded victims
4. death occured before being transported to hospital
5. death occured after receiving treatment in hospital

3.1.5 Preprocessing[9]

Let $F = \{\text{'violations'}, \text{'status'}, \text{'transport'}, \text{'nearby'}\}$ – features to be transformed.

Let S be a set of all available features. $C \subset S \setminus F$ – categorical features to be encoded.

Let E be the space of all possible feature encodings.

Let $g \colon F \hookrightarrow E$ – injective function that applies feature specific encoding.

Let $h \colon C \hookrightarrow \mathbb{N}$ – function that returns number of unique elements for given feature.

Let D be our sample of accidents.

When we apply some function to the feature, we assume that feature is a column of size of all our data, e.g. 'violations' $\in X^{|D|}$, where X is a set of unique violations. This is done to remove multiple for-loop nesting over all data points when describing algorithm.

Preprocessing algorithm

Initialize: $F_g \leftarrow \{\}$
for $x \in F$ **do**
    $F_g \leftarrow F_g \cup g(x)$
end for
$C \leftarrow C \cup F_g$
for $c \in C$ **do**
    $T_c \leftarrow one\_hot\_encoding(c)$, where $T_c = \left\{t_1, \ldots, t_{h(c)}\right\},\ t_j \in \{0,1\}^{|D|},\ j \in \{1, \ldots, h(c)\}$
    $C \leftarrow C \setminus c$
    $C \leftarrow C \cup T_c$
end for

This is a general procedure which happens when we preprocess our data in `download_and_preprocess_data()`. Eventually, we get a dataset with all the features that we want to explore. Some features like 'driver experience' are extracted from nested structure and put in a list because there can be multiple drivers involved in an accident. We also extract percent of women involved in an accident as driver, e.g. if there was a collision between a man driver and a woman driver, then the value would be 0.5. Lastly, we select only those accidents that occured in Moscow. Initial data has some flawed points that need to be removed. That is why we use the following bounding conditions on latitude: from 55.1339600° to 55.9825000°, longitude: 37.1813900° to 37.9545100°, which correspond to Mihnevo (55.1339600°, 37.9545100°) and Zelenograd (55.9825000°, 37.1813900°). Data is saved in `csv` format and is accessible from root of repository. The code is applicable (maybe with minor tweaks) to all other Russian regions. Final dataset contains 54599 entries and 99 features. We leave all the features presented in the initial data, except ones that were difficult to encode and interpret, e.g. 'driver's experience'[10]

[9] https://github.com/isdevnull/cw3/blob/dev/preprocessing.py

[10] We extract it, but don't use it further.

– if there are multiple cars in an accident, it does not make any sense to employ averaging strategy, because driver's experience may vary greatly from one year of driving to 60 years, but it is difficult to differentiate between experienced drivers. One more feature that we leave out is a 'car's model' – there is just too many of them and there is nothing except model's name and we would probably be interested in their characteristics, e.g. safety test results, which would require additional collection of data, so we leave it for another time.

4 EDA

4.1 Initial analysis

First, we decided to explore our data with respect to severity, which is useful for our analysis. We can see

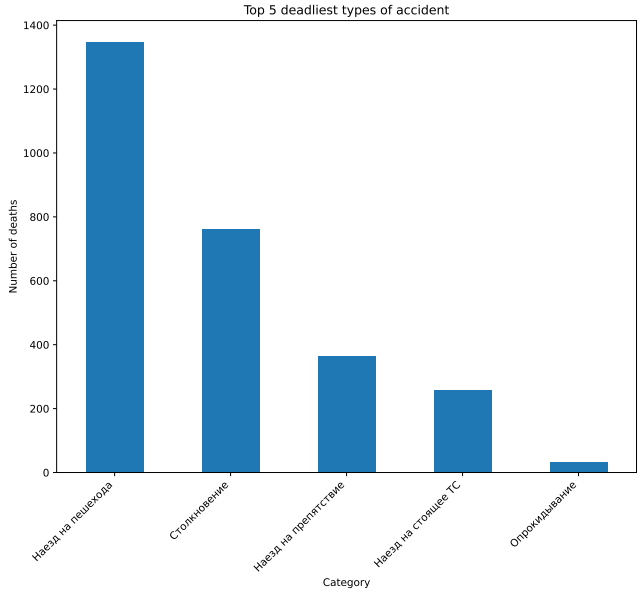

Figure 1: **Top categories by deaths**

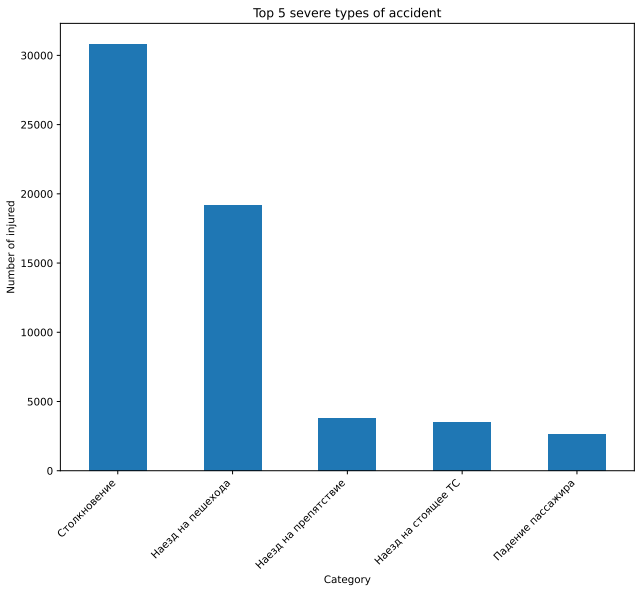

Figure 2: **Top categories by injuries**

from figure 1 that most of the deaths occur when pedestrians are involved, next deadliest are crashes, obstacle encounter, crashes with non-moving vehicles, overturns. Nothing surprising. We get similar results for injuries: Most injuries occur during crashes, pedestrian and obstacle encounters, crashes with non-moving vehicles, falls of passengers. Interestingly, if we count number of deaths with respect to total participants by category of accident, we get the results as shown in table 1. The high mortality rate is not surpising. There are relatively few participants in these types of accidents (less than 130), except obstacle encounter. All of these events are quite similar in terms of that they mostly happen on motorways, where speeds are high and thus crashes are more lethal when they occur. However, as we saw from figure 1 most deaths occur in habited areas.

4.2 Clusterization[11]

Our data possesses geospatial properties, that is why we decided that it would be nice to explore potentially dangerous areas. There are just too many accidents in Moscow, so it may seem that they cover almost all of the streets and motorways and in fact it is absolutely true. However, we can employ density-based clusterization

[11] https://github.com/isdevnull/cw3/blob/dev/clusters.ipynb

Category	P_{death}
Hitting maintenance worker	0.23
Crash into ditch	0.14
Hitting other	0.12
Hitting police officer	0.09
Running over animals	0.09
Obstacle encounter	0.09
Overturn	0.05
Other type of accident	0.05

Table 1: **Top 8 Probability of death with respect to accident**

technique to select only those spots on the map that are most dense in terms of car accidents. To do that we use DBSCAN [3], which can select clusters of complex form, depends only on a few hyperparameters which are easily interpretable.

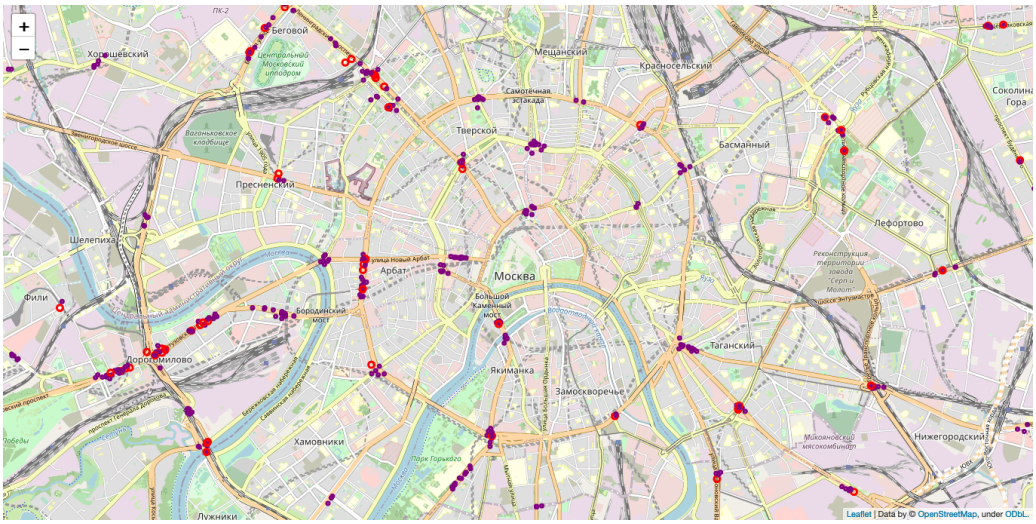

Figure 3: Clusterizaion with U=$\frac{0.1}{6371}$, N=15; red points – deadly accidents, purple – heavy severe

These are neighbourhood of a point – U and number of members – N to be considered as core point. Due to Earth being spherical we use Haversine distance and transform our geospatial data to radians, thus neighbourhood U in our case is distance between multiple accidents and core samples are defined as those that have N accidents around it, scattered in time. When DBSCAN selects M clusters, we firstly delete noise points, then we do not distinguish between clusters, because we are only interested, in general, how problematic regions are distributed. We color accidents according to their severity. The larger the U hyperparameter the bigger the problematic regions and the more spreaded they are.

4.3 Time Series Analysis

Because we want to predict severity and it is a categorical variable, we assign label 0 to cases without significant injuries and 1 otherwise. First of all we examine the behavior of the variable. As we can see the number of severe car accidents declines as shown on figure 5. However as it was mentioned above severity combines two

features. Therefore we plot number of injuries and deaths per month as well. As we compare two figures 6 and 8, it is clear that the downward trend of the number of severe cases can be attributed the the decline of number of deaths. The graph for injuries(6) heavily fluctuates around its mean, whereas the girth depicting number of deaths declines. If we review this graph separately we may notice that beside downward trend it also has a yearly seasonality. We visualize the behaviour of number of deaths (figure 8) and compare it to the number of car accidents throughout a day. The analysis shows (figure 7) that the number of accidents peaks at a daytime, however the most risky time for a person is night, as it is four times more likely to be killed or injured from midnight to 7 a.m. Based on that information we construct additional features and estimate their potential to be useful predictors. Correlation of the newly created predictors is shown below:

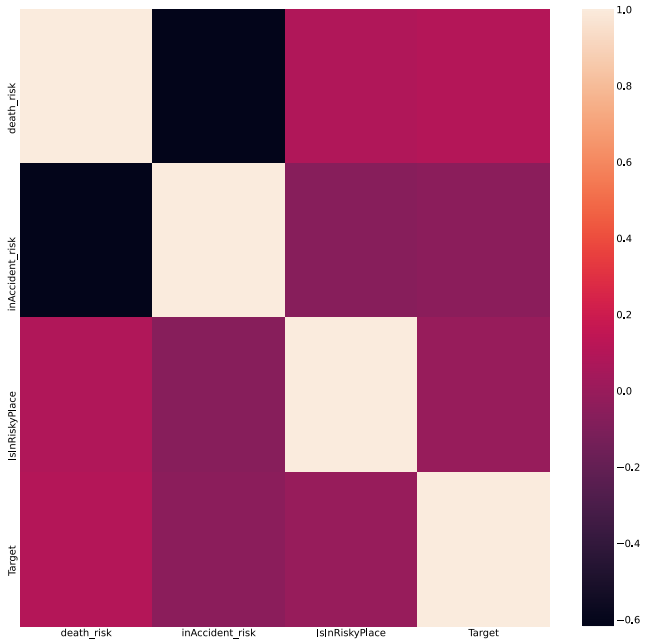

Figure 4: **Correlation of embeddings**

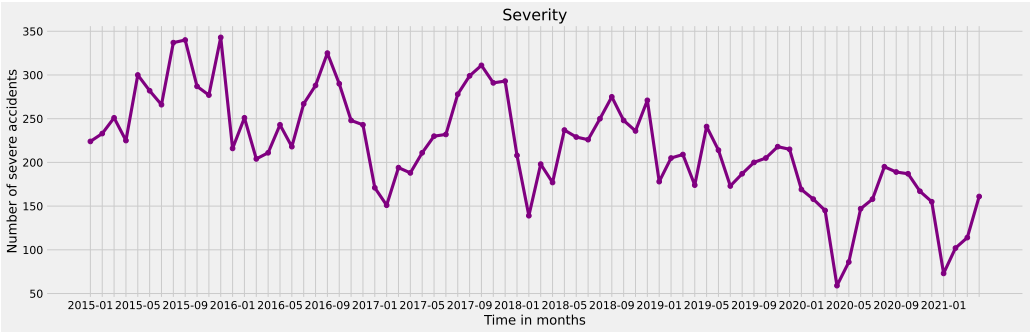

Figure 5: **Severity from 01.2015 till 04.2021**

5 Model

5.1 Time Series

To analyze time series data we use three models to predict the trajectory of the number of accidents with deaths. We split the series into 'Train' and 'Test' putting off the last year for testing. For time series analysis we fit

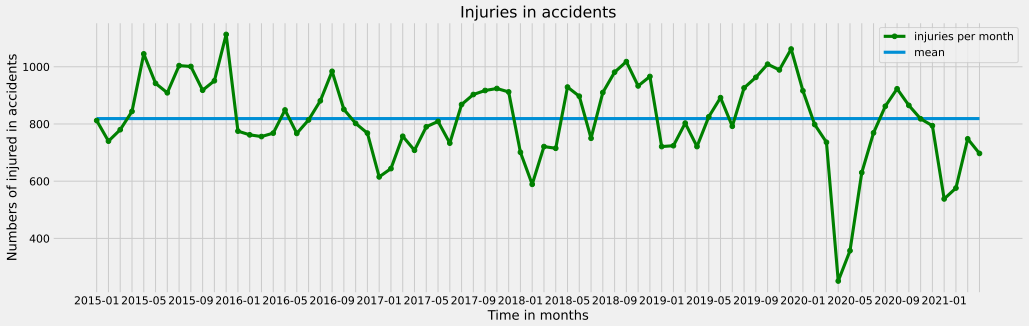

Figure 6: **Number of injuries from 01.2015 till 04.2021**

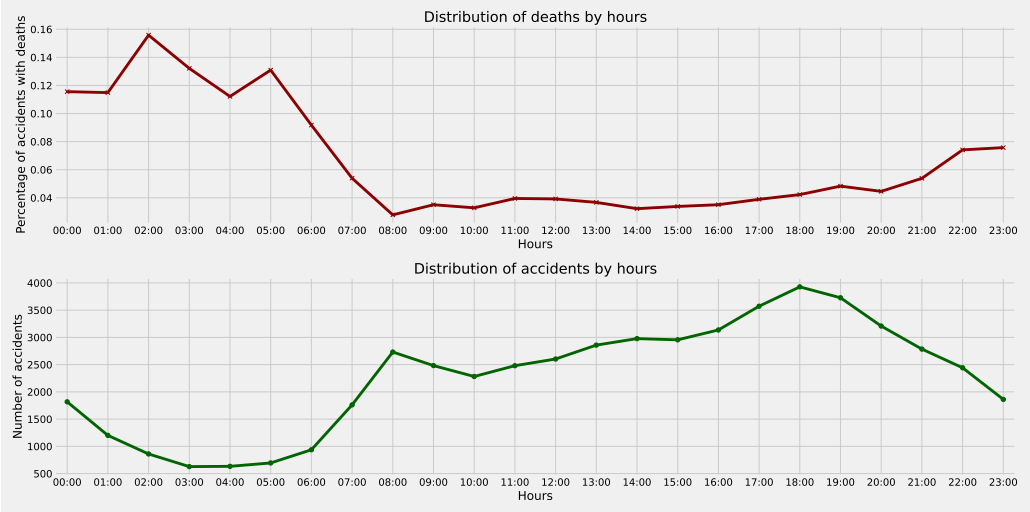

Figure 7: **Distribution of deaths and accidents by daytime**

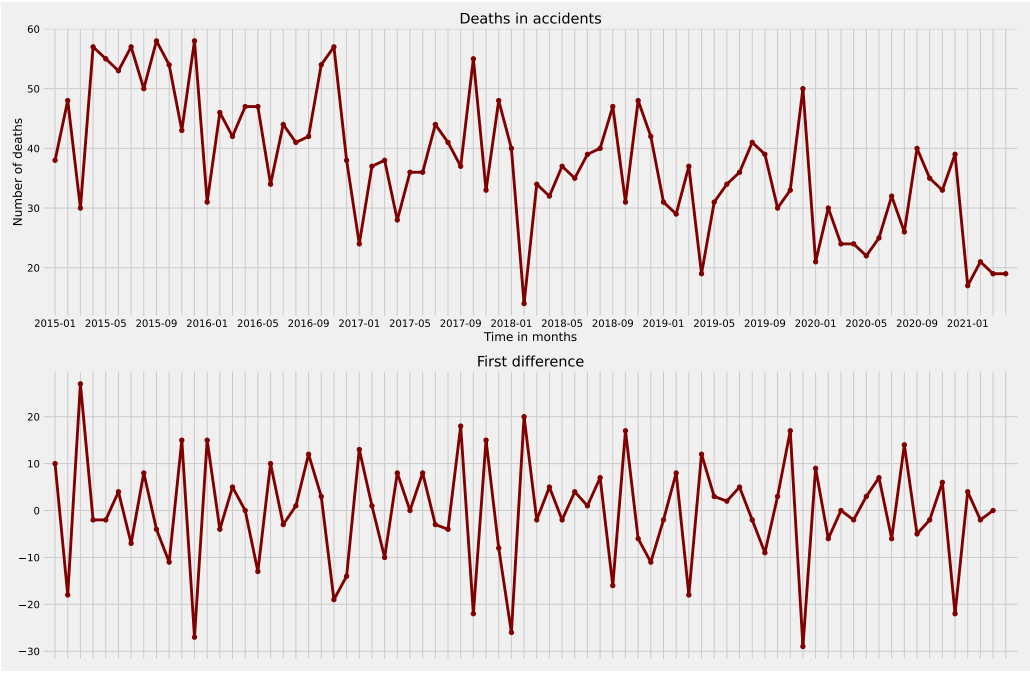

Figure 8: **Number of deaths from 01.2015 till 04.2021**

ARIMA model [SARIMAX(0, 1, 1)(1, 0, 1, 12)] with its AutoARIMA implementation, TBATS with Box-Cox transformations and year seasonality and Prophet [4] with additive seasonality, linear growth and MCMC samples using their implementation in `sktime`. In order to measure the error of prediction we use MAPE (Mean average percentage error). The choice of the metric can be attributed to the need to measure not the absolute value of the error, but the normed one, as these results can be interpreted better. The results of the models are shown in the table.

	MAPE
ARIMA	0.2697
TBATS	0.2951
Prophet	0.2271

Table 2: **Mean Average Percentage Error**

We visualize the predictions of each model as shown on figure 9. As it turns out the best model is Prophet with MAPE equal to 0.2271.

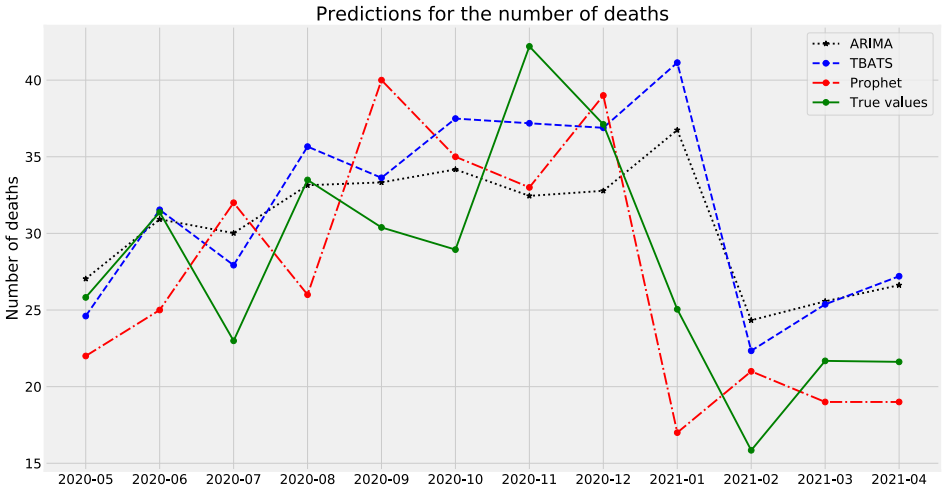

Figure 9: **Time Series Death Predictions**

5.2 Imbalanced Classification

In this part we carefully examine the approach to imbalanced classification used in this research. In order to measure the accuracy of prediction we choose Cohen-Kappa coefficient [5] and Matthews Correlation Coefficient [6, 7, 8]. We also use AUC-Pr to measure the quality of the model. The reason we choose these metrics instead of usual accuracy is precisely the imbalance we have. In that case usual accuracy happens to be skewed towards majority class and does not accurately measure the quality of the result. Moreover, we separately measure precision and recall of two classes as well. First of all we focus our attention on classical methods and use logistic regression and boosting methods. As hyperparameters we select class weights, significantly downsizing the majority class the results of two methods are shown on figures 10, 11.

In order to meaningfully estimate the results of the two models we look at probabilities the models assign to each class for every object and then compute metrics multiple times while changing the binarization threshold. We can see that two models give comparable results with boosting performing slightly better. The results can be interpreted as follows. As it was stated before two classes are heavily imbalanced, and they do overlap as well. That highly influences precision of a minority class. It is very low. The only pick of precision occurring while the threshold is low is attributed to the tiny sample of minority class objects that the models are quite sure about, however this result cannot be interpreted as successful one due to recall being very low in this case. Low precision of minority class consequently influences metrics for imbalanced classification that we use. It is clear from the graphs, that both models are unable to give meaningful accuracy. As authors think, it happens largely to the overlap of data. That means that existing predictors are insufficient to accurately separate to classes. Meaning that we actually really don't have other valuable features, that may help us to predict the severity better.

In attempt to tackle that problem we fit models, specifically designed for imbalanced datasets. Their results can be seen in the table.

	Cohen-Kappa	**MCC**	**AUC-Pr**	**Precision(Majority)**	**Precision(Minority)**	**Recall(Majority)**	**Recall(Minority)**
One-Class SVM [10]	-0.025	-0.05	0.044	0.938	0.036	0.491	0.371
Isolation Forest [9]	**0.054**	**0.054**	**0.054**	**0.954**	**0.094**	**0.039**	0.122
Local Outlier Factor [11]	0.001	0.005	**0.054**	0.953	0.050	0.314	**0.696**

Table 3: **Imbalanced Classification Metrics**

As we can see these methods perform worse, with Isolation Forest giving slightly better results. The reason for that may be the overlap as well. And given the fact that all the methods are based on either drawing a hyperplane in object space or counting the distance they cannot discriminate objects of two classes as those objects are very close to each other.

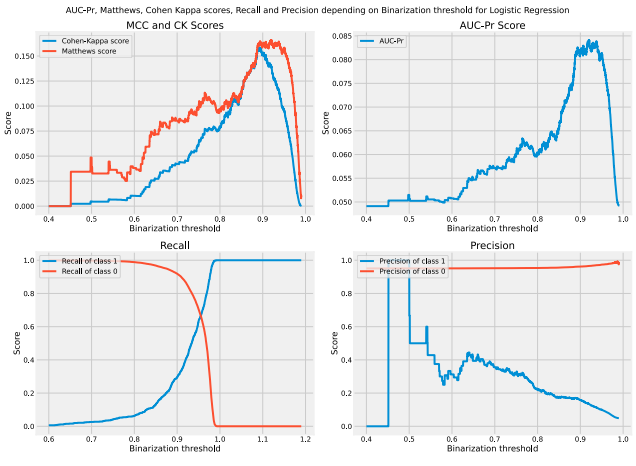

Figure 10: **Classification Metrics Logistic Regression**

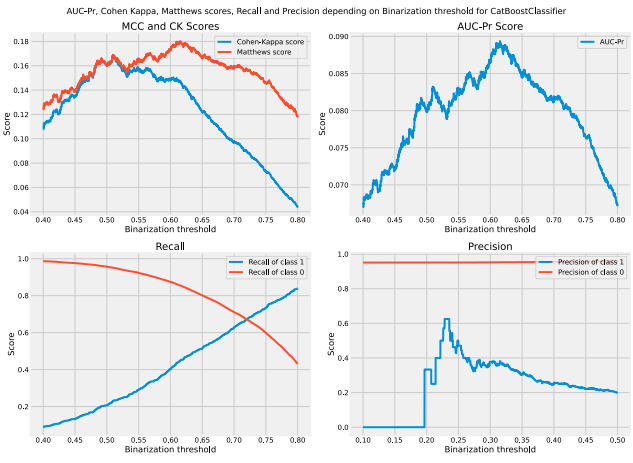

Figure 11: **Classification Metrics Catboost**

6 Conclusion

To conclude the whole research we can say, that overlapping data and imbalance significantly impeded the model construction. Now let us review the dataset and draw meaningful summary of that work.

Firstly, exploratory data analysis clearly shows that, although the number of car crashes is at its height during the day, one is less likely to end up being killed at noon. However, the chance of a tragic outcome increases almost fourfold at night. Clustering car accidents also allows to obtain information on the most risky places. Those places turn out to be junctions or motorways with a heavy traffic.

Secondly, the model for predicting the number of deaths in car accidents was constructed, emulating the yearly seasonality and clear downward trend. Based on comparison the best model turned out to be Prophet with Bayesian inference and MCMC simulations.

Thirdly, as we can see the existing data is insufficient to determine the outcome of a car accident. It does not mean, that road or weather conditions have no influence on the outcome of a crash, but that their impact is minor. At least there are other factors that are relevant and are not present in the dataset. And that is actually true from the practical point of view. The dataset has no information concerning the type of a vehicle, its conditions and safety equipment. However, these things are highly important as they determine the extent to which driver is protected. Moreover, we examined only the given features, but there is an option to construct some new features out of them. For example one could create some combinations of features or use kernels to perform a non-linear transformation of given information.

So, based on the existing information it is impossible to meaningfully predict the outcome of a crash. Therefore another research is needed to extract additional information and build a better model .

References

[1] Matts-Ake Belina, Per Tillgrena, and Evert Vedungd, Vision Zero – a road safety policy innovation, *International Journal of Injury Control and Safety Promotion*, **19**, No. 2, 171-179, 2012.

[2] Проект «Карта ДТП»

[3] Ester, M., H. P. Kriegel, J. Sander, and X. Xu, A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, 226-231, 1996.

[4] Sean J. Taylor, Benjamin Letham, Forecasting at scale, *The American Statistician*, (2018).

[5] J. Cohen, A coefficient of agreement for nominal scales, *Educational and Psychological measurement*, **20**, No. 1, (1960).

[6] Matthews B. W., Comparison of the predicted and observed secondary structure of T4 phage lysozyme, *Biochimica et Biophysica Acta (BBA) - Protein Structure*, **405**, 442–451, (1975).

[7] P. Baldi, S. Brunak, Y. Chauvin, C. A. Andersen, H. Nielsen, Assessing the accuracy of prediction algorithms for classification: an overview, *Bioinformatics*, **16**, No. 5, 412-424, 2000.

[8] D. Chicco, M. J. Warrens and G. Jurman, The Matthews Correlation Coefficient (MCC) is More Informative Than Cohen's Kappa and Brier Score in Binary Classification Assessment, *IEEE Access*, **9**, 78368-78381, 2021.

[9] Fei Tony Liu, Kai Ming Ting, Zhi-Hua Zhou, Isolation Forest, *Data Mining*, 2008.

[10] Bernhard Scholkopf, Robert Williamson, Alex Smola, John Shawe-Taylort, John Platt, Support Vector Method for Novelty Detection, *NIPS*, 1999.

[11] Breunig, M. M., Kriegel, H. P., Ng, R. T., Sander, J, LOF: identifying density-based local outliers, *ACM sigmod record*, 2000.