*Showcase of database analytics on car accidents dataset*

Matej Poljak  
*Faculty of Management Science and Informatics*  
*University of Žilina*Žilina, Slovakia  
poljak@stud.uniza.sk

*Abstract*— The number of cars has been rapidly growing in recent years and so has the number of related accidents. Given the availability of a public resource providing this data, it would be interesting to analyze it from various aspects. During this work we quantify the impact of alcohol usage on car accidents, considering factors such as accident frequency, time of day, regions and associated damage. The results may be distorted because gathered data does not seem to correspond to reality always. The main goal of this paper involves suggesting database analytic tools (specifically aggregate and analytic functions) available on Oracle platform which provide us with an effective and easy extraction of information from vast amounts of data. After that we show optimalizations of created queries using database indexes. Eventually, this study demonstrates the power of these tools, as well as the importance of proper data gathering. The results can motivate national institutions to focus more on precise data gathering for matters such as car accidents, as significant information can be gained from such data to implement smarter measures that contribute to a better world.

Keywords— analytics, alcohol usage, car accidents, database

# Introduction

This work is trying to emphasize the importance of gathering and processing data while we live in era of data explosion. Spread of the Internet, social networks, Internet of Things, cloud-computing, etc. This all comes with the vast amount of data generated. But there are also areas that are not still monitored well or at all, but they should. The reason we should care about it is that we can extract from them some interesting and important information which can help us to understand some phenomenon, process and (ideally) based on that create better solutions if possible. As a good example of consistent data gathering is Czech police, which reveal car accidents data every month on their web [1] and are free for everyone. These data are going to be used as an example of data processing using Oracle database analytic tools. Specifically, this work focuses on alcohol impact of car accidents from various points of view. It is possible to observe this while information about alcohol usage is provided within stored data.

# Dataset overview

## Preparing dataset

First, we need to download data from [1] in zip format. We will focus on the year 2024 only – that means we must download twelve structured files. These files contain IDs from many lookup tables. Values of lookup tables are specified in file *Položky formuláře – data* also available on [1]. Then we can create database tables. Final structure is shown in Fig. 1. There are many references to database tables that serve as lookup tables. For our work, we will use just these tables: (since the names of tables and their attributes are in the Slovak language, we will use aliases throughout this paper to enhance reader comprehension)

* Table *CR\_NEHODY* – this is the main table that stores all information about car accidents and IDs to other lookup tables. We will refer to this table as *acc* (shortly for accident).
* Table *CR\_KRAJE* – is a lookup table for regions in Czech Republic. We will refer to this table as *regions*.
* Table *CR\_PRITOMNOST\_ALKO* – is a lookup table of observations of alcohol level in the blood of person that was responsible for an accident. We will refer to this table as *alco*.

Downloaded zip file from [1] contains *xls* file with records. Records can be imported to the database by creating formulas that construct insert statements.

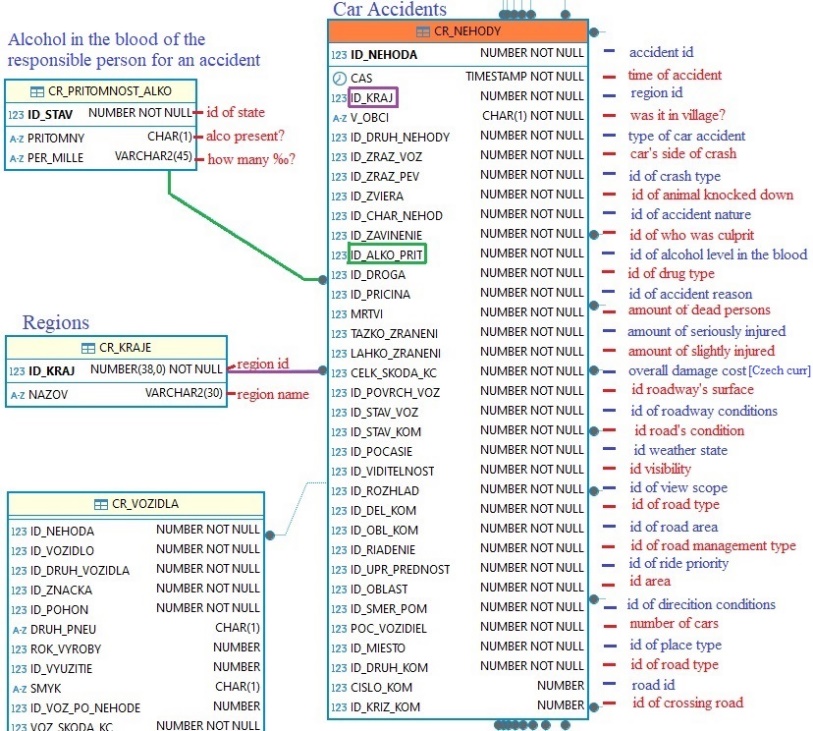


Figure 1 - Structure of database tables

## Understanding possible cases of alcohol usage

We already know that *acc* table has a relation to *alco* table. Now, we could get insight into what alcohol cases were observed when an accident occurred. Results of select query are shown in Fig. 2. We see that recordings can be negative on the presence of alcohol or positive with different amounts of per mills. There can also be records with refusion of alcohol observation for an unknown reason. There are also cases when it was not observed at all.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 2 - Presence of alcohol during an accident (script)

# Basic statistics

With knowledge from the previous paragraph, we can dive into analysis of our dataset.

## All accidents and alcohol-related accidents

Before we start doing more in-depth analysis, we should get a general overview of accidents in 2024, as well as the number of accidents which were somehow related to alcohol usage. Script with results is shown in Fig. 3. In the first select of UNION statement we query for all accidents in 2024 with any per mill amount of alcohol found in the blood. We sum all cars that were related to those accidents also. In the second select of UNION statement we query for all accidents in 2024. Results of UNION statement are then grouped to one row using analytic function lag (assigns each row its predecessor) and filtering proper row. We see that only 4475 out of 91211 accidents had relation to alcohol usage, what is approximately 5%. 6808 crashed cars were found in accidents with alcohol involvement, what is also 5% of all crashed cars (149 208). We might have expected a higher number of accidents related to alcohol usage.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 3 -Accidents for each possible state from alco table (script)

## Observations for all possible states of alco table

Let us observe the number of accidents for each case of *alco* table to get a better idea of what is going on. Results from Fig. 4 reveal that 37 572 accidents, what is 41% of all accidents, were without any observation of alcohol involvement what is interesting, and we do not know the real reason for that. In most cases, which is 53% of accidents, the presence alcohol was not confirmed as shown in Chart 1. In the query from Fig. 4 we compute number of cases for each alcohol presence in the blood or absence of all three types {N,X,O}. Relative proportions of each state are also computed to understand better dominant cases.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 4 - Ratio of alcohol-accidents to all accidents (script)

Chart 1 - States of alcohol observation in the blood during an accident

Another interesting fact is that if there was an accident with confirmed alcohol, then the probability of measuring at least 1.5 per mills in the blood is greater than 60%. (Chart 2) This assumption was taken by extracting all result rows with positive presence of alcohol in the blood.

Chart 2 - Amount of alcohol per mill in the blood

# Analysis based on time

Now that we understand work with these database tables, we can conduct wider analysis. We will start with time analysis of car accidents for the year 2024.

## Monthly alcohol in-accident involvement

In the previous chapter, we have found out that majority of accidents end up with any alcohol involvement or it was not observed. We should investigate whether this behavior was the same throughout the entire year or if it has been changing. We are going to focus just on four alcohol-type states: alcohol present, not present, not observed, refused. We start by grouping records by extracting the month of accident occurrence. While we want to pivot counts of wanted alcohol-types states for each month, we can create sum expressions with CASE clauses to count only cases when an accident contains desired alcohol-type state. Besides that, we compute overall counts of cases for each month to compute relative proportions of alcohol-types later. The whole script is shown in Fig. 5. Results are visualized from two perspectives in Chart 3 and 4. From Chart 3 we can see that proportions are similar throughout the entire year. Chart 4 suggests that most alcohol-present accidents were recorded in the summer months July and August. To confirm this observation and see more detailed numbers, we can look at Chart 5, which visualizes numbers of column *alco\_yes* from script in Fig.5. We can assume that this could be related to the time of vacation when people are more prone to drink alcohol and forget to be responsible.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 5 - Monthly ratios of alcohol involvement in accidents (script)

Chart 3 - Monthly ratio of states of alcohol observations during accidents (percentual)

Chart 4 - Monthly ratio of states of alcohol observations during accidents (amounts)

Chart 5 - Number of accidents with found alcohol per month

## Amount of accidents based on the hour of day

Next, we can take a closer look at which hours of the day are most significant regarding accident frequency. We must group records by hour of the day. This can be done using the *EXTRACT* function where we define that we want to get hour of day (script is shown on Fig. 5). We can see on Chart 5 that most frequent hours are from 18 to 20 PM when people tend to meet or have more free time. We can also notice that there is a high number of accidents before midnight. This can be related to when people come home from some event where they have drunk.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 6 - Number of accidents for each hour of day (script)

Chart 6 - Number of accidents for each hour of day

## Amount of accidents for each day in the week

Aside from the typical hours associated with frequent accidents, it is also important to observe the development of accident frequency across the days of the week, based on data for the entire year 2024. Query (Fig. 7) is simple but can tell us very interesting information. That is the power of database analytic tools. We group data by converting timestamp to string in format of day of week using *TO\_CHAR* function with parameter ‘Day’. Then, we can simply compute a count of cases that occur on their corresponding day of week. The number of accidents during the weekend is the highest, but especially on Saturday. We can assume that more people are prone to drinking more on Saturday because they have one extra day to get fit before they go to work.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 7 – Number of accidents for each day of the week (script)

Chart 7 – Number of accidents for each day of the week

# Analysis based on regions

We have already provided insights into the most accident-prone hours of the day, days of the week, and months, along with presenting their behavioral trends. Now moving on analysis on region level to find out which regions are related with most accidents related to alcohol usage.

## Accidents on region level

Firstly, we will compute the accident amounts for each region using the complete dataset from 2024. Script is shown in Fig. 7. We join accidents with corresponding regions from *regions* table and then group them by region identifier. After this, we calculate the number of accidents for each region.

Results shown in Chart 8 indicate that region Středočeský kraj experienced far more accidents than any other, more specifically over 650 accidents. Three following regions, Moravskoslezský kraj, Prague and Jihomoravský kraj, had over 400 accidents with alcohol in the blood in 2024.

We must bear in mind that each region has a different number of residents; therefore, these figures do not imply that regions in TOP positions have the highest ratios of accidents relative to their resident populations.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 8 - accidents of each region (script)

Chart 8 - Number of accidents per region

## Regions with consistently high accident occurrences in monthly rankings

Previously, we have analyzed accidents per region for the whole year. Now we want to know which regions consistently appeared among TOP 3 regions with highest numbers per month. We also want to know the exact numbers of region appearances in TOP 3.

Firstly, we compute the number of accidents for each group defined by region ID and month of the year. Then, we assign a ranking to every record based on the number of accidents. This ranking is performed independently for each month. Subsequently, we determine how many times each region appeared with a ranking lower than or equal to 3 (meaning within the TOP 3). After that, we apply the analytic function *RANK* again, but this time on the number of occurrences in the TOP 3 across all months. The resulting select expression is shown in Fig. 9.

Now we see from looking at Chart 9 that the first three regions are different from the first three regions in Chart 8. Regions that have occurred most times in TOP 3 are Karlovarský kraj (11 times) and Vysočina (10 times). These results are absolute, not proportional.

It is important to note that the result of this query could be empty if different regions experienced the highest number of accidents each month. However, this was not our case. If it was, we would simply modify the constant against which the rank value is compared or remove WHERE condition entirely.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 9 - Regions appearing most times in TOP 3 monthly accidents count (script)

Chart 9 - Regions appearing most times in TOP 3 monthly accidents count

## Weighted average ratio of alcohol-involved accidents per region and month

We move from absolute to relative numbers. As shown in Fig. 10, we compute the weighted average of monthly accidents counts for each region. The number of days serve as the weight. Assigning IDs is done to accomplish having results of both united select results in one row. This technique was used in the third chapter also. Weights representing days of specific month are computed as composing the last day of month which is then extracted from the resulting value. Finally, ranking is assigned to each result by ordering output based on weighted average ratio.

A screenshot of a computer screen

AI-generated content may be incorrect.

Figure 10 - Weighted averages of monthly alcohol-related accident ratios (script)

Chart 10 - Weighted averages of monthly alcohol-related accident ratios

## TOP 3 hours with most alcohol-accidents for each region

In the previous chapter we created a query to find out the most frequent hours of the day regarding accidents with alcohol involvement. We found out that most frequent hours are 18, 19, 20 and 23. Let us now break this analysis into smaller units – regions. We will observe TOP 3 most frequent hours for each region of Czech Republic and see whether these hours are most frequent almost everywhere or not.

We create groups of accidents by hour of day and region ID to find out the wanted number of accidents. Then we use the analytic function *ROW\_NUMBER* to assign rankings to hours (based on descending ordering of the number of accidents) within partition defined by region ID. We use *ROW\_NUMBER* function instead of *RANK* function, because we will create exactly three series for TOP 3 hours. Lastly, we filter output to get just results with ranking better or equal to constant three and then we order results by ranking to group numbers for TOP 1, 2 and 3 hours. We also then order results by region ID to get numbers of corresponding region on the same position in every TOP time series.

To understand the visualization of Chart 11, we can take an example of Prague. We can see that in Prague, most accidents occur during 17-th hour, the second most frequent time was 23 PM and third was 01 AM. Judging by the results from Chart 11, we cannot say that TOP 1 most accidents occur at the same hour in each region. It depends on the region. On the other hand, what we can say is that generally, the most frequent accident hours are surely after 16-th hour.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 11 - TOP 3 hours with most alcohol-type accidents for each region (script)

Chart 11 - TOP 3 hours with most alcohol-type accidents for each region

# Analysis of damages

The final topic concerns the damage caused by accidents in which alcohol is present at any positive per mill level.

## Average number of vehicles involved in an accident

To begin with, we can review the statistics on the average number of cars involved in accidents with alcohol found alcohol.

But before that, we can simply get middle value of the number of vehicles per accident. Table of accidents provides us with number of cars involved in a single accident, so we can apply aggregate function *MEDIAN* over this table (Fig. 12) to get the middle value. From the dataset, we observed a median of two vehicles per accident. A single-vehicle accident is also a possible scenario, particularly in cases involving a collision with a pedestrian, an animal (such as forest game) or fixed object.

A screen shot of a computer

AI-generated content may be incorrect.

Figure 12 - Median of vehicles per accident (script)

Let us now compute the weighted average of cars involved. As Fig. 13 suggests, we use here a common table expression (CTE) named as *vehicles\_amount* for better readability. In that *vehicles\_amount* we group records by vehicles involved in an accident so we can compute the number of such accidents. We also want to get that number of vehicles involved, because this value will be used as the weight for computing the average with weights in the next part. For our curiosity we add aggregate function *MAX* to find out maximum vehicles involved in a single accident.

Results from Fig. 13 indicate that, on average, 1.5 vehicles are involved in an accident. Besides that, there was at least one accident in which had crashed 9 cars. This reveals how dangerous drinking alcohol can be before entering the road. For better understanding of how many accidents occurred for each possible state of the number of vehicles per accident, we plot histogram of these values. (Chart 12) We observe that most alcohol-related accidents involved one or two vehicles, with single-vehicle incidents being the most prevalent in over 2500 cases.

A screen shot of a computer code

AI-generated content may be incorrect.

Figure 13 - Weighted average number of vehicles in alcohol-related accidents (script)

Chart 12 - Number of accidents for each possible number of vehicles per accident

## 2.5-month moving median of injured per half-month

In the *acc* table, information regarding the number of slightly and seriously injured individuals is stored in two separate columns. For this analysis, we aim to determine the trend and seasonality of a time series composed of the total number of injured persons, which is derived by summing both slightly and seriously injured counts.

To make it more interesting we will compose time series with frequency equal to half-month. This means that we must sum the number of injuries into half-month intervals. We will divide the entire script into two parts. In the first (Fig. 14), we will focus on creating half-month data, which we will use in the second part (Fig. 15).

In the first part of this process, we begin by creating records with numbers 1-12 to represent monthly numbers. These are used in CTE named *months*, where we compose dates of months beginnings using the *TO\_DATE* function with the corresponding month number. As the next step, we create another CTE named *halfmonths* containing half-month dates. We take all dates from the *months* CTE and union them with the newly created half-month dates. These half-month dates are generated by determining half the number of days between the beginning and the end of a specific month (plus one day for the midpoint) and then adding this resulting half-month days count to the beginning of the month. In the last step of the first part, we create CTE named *halfmon\_data* where (in the inner select) we join *acc* table with *halfmonths* on their corresponding beginning of the month. This way, we get two result rows for each accident: one with the date of the beginning of the month and the other with the date in the middle of the month. We want to retrieve only one of these two resulting rows. To retrieve only one of these two resulting rows, we introduce an *hms\_priority* column into the inner select query. This column assigns priority by ranking both rows (partitioning is done for pairs). Specifically, the *ROW\_NUMBER* function, utilizing a *CASE* statement in its ordering clause, assigns the highest priority to the half-month date that is chronologically closest to and immediately following the accident's occurrence time. After that, we can filter rows to get only those which have the highest priority (*hms\_priority*=1) and then compute sums of injuries for all groups defined by their half-month.

In the second part of the script, we utilize the *halfmon\_data* CTE. We note that the *halfmon\_data* CTE contains a distinct row for each half-month, meaning the half-month value is unique within *halfmon\_data*. We aim to union each half-month with its corresponding number of injuries and with the number of injuries from the four previous half-months. Retrieving these previous numbers is achieved by using the analytic function *LAG*, within which we define the constant for the N-th previous lag and specify the ordering of rows by half-month date value before picking the lags. We do this, because we compute 2.5-month moving median, that means we need five half-months (specific half-month + 4 previous).

The resulting time series is visualized in Chart 13. We can observe an ascending trend from the beginning of the year 2024 up to mid-September, after which the trend changes to descending. No seasonality is apparent in this single year's data; however, analyzing data over multiple years could reveal signs of cyclicality. An unambiguous observation is that most injuries during accidents with alcohol involvement occurred during the summer months.

A screenshot of a computer code

AI-generated content may be incorrect.

Figure 14 - Composing half-month data for computing moving median of injured (script)

A screenshot of a computer

AI-generated content may be incorrect.

Figure 15 - Computing windowed median on grouped half-month numbers of injuries (script)

Chart 13 – 2.5-month moving median of injured per half-month

## Contribution of alcohol-related accidents to total damage in the TOP 1000 most costly accidents

Lastly, we compute the percentage of total damage attributable to alcohol-related accidents among the top 1000 most costly accidents. We are curious whether alcohol contribution has great consequences or not in those 1000 cases. For better understanding of how we are going to construct select queries, we visualize it first using Fig. 16. We must create two select queries, *damage\_alco* and *damage\_overall*, from which we will then compute wanted ratio.

A screenshot of a computer program

AI-generated content may be incorrect.

Figure 16 - Template for query of contribution of alcohol-related accident damages (script)

Both queries (Fig. 17) are straightforward. Starting with the *damage\_overall* query, we assign a ranking to all accidents based on the damage caused by the accident (using the value of the *acc.celk\_skoda\_kc* column). From these results, we select all rows with a ranking lower than or equal to a constant value of 1000. The *damage\_alco* query is constructed similarly, but when extracting the resulting rows, we add a filtering *WHERE* condition that specifies accidents must have some alcohol involvement. Finally, we compute the proportion of the results from these two queries and transform it into a percentage format.

As the results (Chart 14) suggest, alcohol-related accidents account for only a 4% share of the damage caused by the top 1000 most costly accidents.

A screenshot of a computer screen

AI-generated content may be incorrect.

Figure 17 - contribution of alcohol-related accident damages among TOP 1000 (script)

Chart 14 - Proportions of TOP 1000 most costly damage

# Optimization of queries

While we work almost the whole time joining *acc* table with *alco* table, it could be desirable to create an index on a foreign key, which by default does not exist. It could be a bitmapping index because the number of states of lookup table *alco* is finite, more specifically, there are only 8 states what is highly under 1% while the number of records is over 90 000 as observed in the third chapter.

Similarly, we could try to create an index on foreign key referencing to *regions* table.

Another optimization could be creating b-tree index on the timestamp column *cas* in *acc* table because we use this column in *WHERE* clause all the time. **Here is question whether to create more functional indexes or basic column index**.

# Conclusion

**TODO**

We have carried out analysis of car accidents dataset from several points of view: overall frequency, time, region and damage. We can say that the results are surprising, because we would say that alcohol has bigger impact on accident consequences regarding the number of accidents or total damage caused by them. We can declare this, because we aimed in some tasks on relative influence and not absolute one. As observed in chapter 3, the number of accidents where the presence of alcohol was not observed was over 40%, which is a huge number. Results might look different if alcohol involvement had been measured. According to [2], Czech Republic had over 10.91 million residents at the end of 2024. Judging by our numbers of accidents with alcohol and damages of alcohol-related accidents, we can say that residents of Czech Republic are relatively careful and responsible when it comes to drinking alcohol and attending roads.

##### References

1. Policie České republiky, "Statistika nehodovosti," *Policie České republiky*, [Online]. Available: <https://policie.gov.cz/clanek/statistika-nehodovosti-900835.aspx>. [Accessed: July 8, 2025].
2. Czech Statistical Office, "Population estimates, structure, and projection," *Czech Statistical Office*, May 16, 2025. [Online]. Available: <https://csu.gov.cz/population-estimates-structure-and-projection>. [Accessed: July 8, 2025]