Container vessels emission

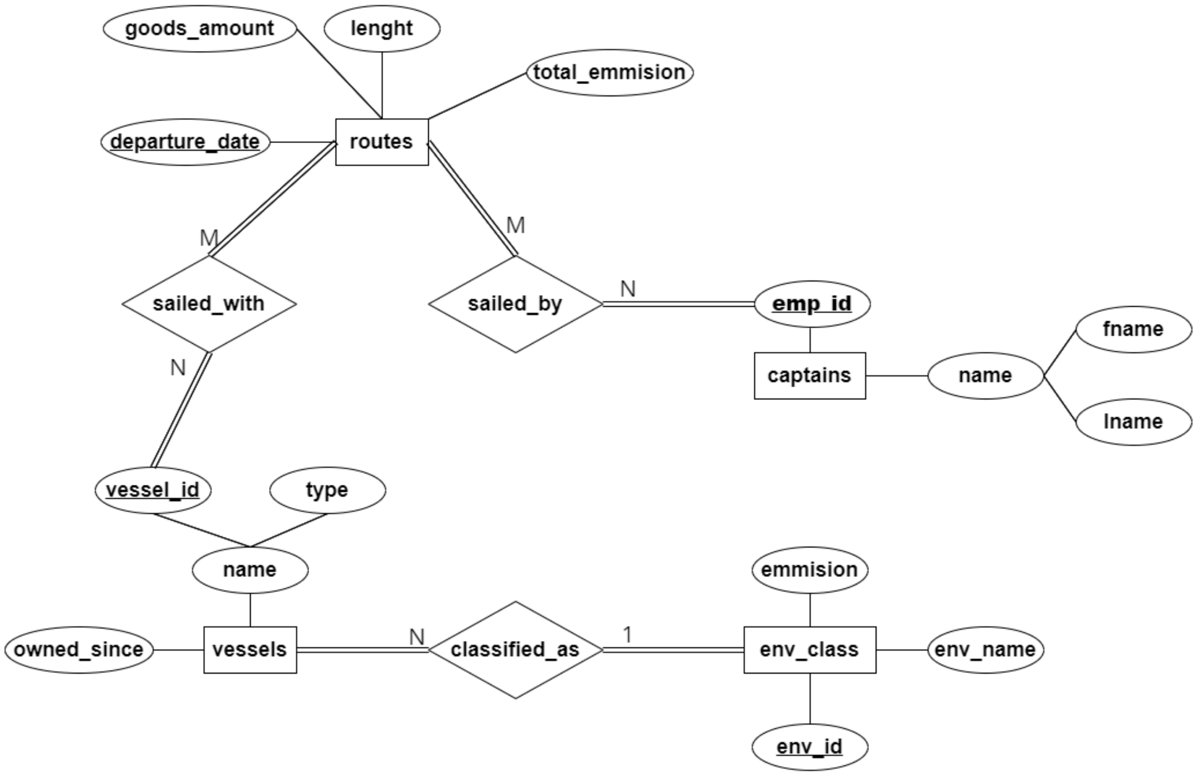
Alexander Husted

February 2023

1 Introduction

The document will take point in a demo SQL database I created. The main focus of the database (fig. 1) will be on the vessels and their environmental impact. Throughout the document, I will answer specific problems using a combination of SQL and R to create an impression of my current capabilities in the different fields.

Mostly you'll be able to find the code under the problem as an answer, but longer code is stored as an appendix. The createDatabase script is the longest and can be found on https://github.com/Husted42/code/blob/main/projects/containerTransport/SQLscrips/createDB.sql Everything is thoroughly tested and runs as expected.



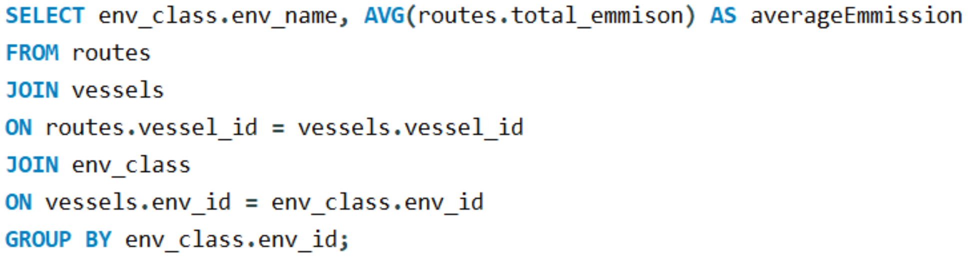
Figur 1: DEMO database

2 SQL

This chapter will resolve problems using simple SQL queries and show a table as an output. The five main focus points are functions, nestedQueries, unions, wildcards, and joins.

Find the average total emission from each environmental class.

env_name	averageEmmission
A	34330.0833
B	49518.6000
C	59295.0000
D	74855.0000

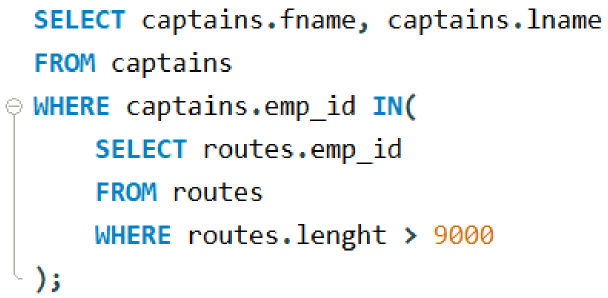


```
SELECT env_class.env_name, AVG(routes.total_emmison) AS averageEmmission
FROM routes
JOIN vessels
ON routes.vessel_id = vessels.vessel_id
JOIN env_class
ON vessels.env_id = env_class.env_id
GROUP BY env_class.env_id;
```

As expected, the better the environmental class, the lesser the average total emission.

Find all the captains that have sailed longer than 9000 km. in a single route.

fname	lname
Archibald	Haddock
James	Hook
Bartholomew	Roberts



```
SELECT captains.fname, captains.lname
FROM captains
WHERE captains.emp_id IN(
    SELECT routes.emp_id
    FROM routes
    WHERE routes.lenght > 9000
);
```

Out of a total of 5 captains, 3 have experience with sailing routes longer than 9.000 kilometers.

Find all of the captains and vessels names.

fname	lname
Archibald	Haddock
James	Hook
Ferdinand	Magellan
Bartholomew	Roberts
Ferdinand	Nelson
1	KFM Gold
2	KFM Silver
3	Zode Silver
4	MST Gold
5	MST Bronze
6	LDS Gold
7	KFM Silver
8	Zode Gold
9	KFM Silver
10	MST Bronze

```sql
SELECT captains.fname, captains.lname
FROM captains
UNION
SELECT vessels.vessel_id, vessels.type
FROM vessels;
```

For both the vessels and the captains, their names contain two different attributes. For the captains, it's fname and lname, and for vessels, it's vessel_id and type.

Find all vessels bought in June and July.

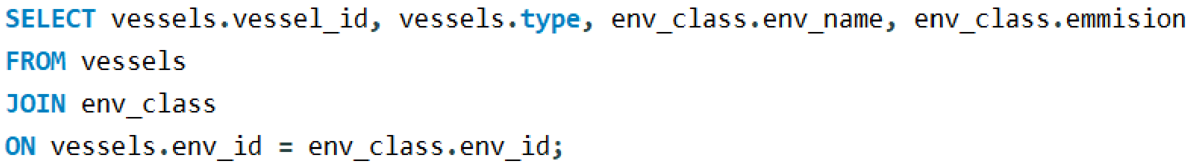
vessel_id	type	owned_since	env_id
3	Zode Silver	1994-06-28	1
4	MST Gold	1995-07-30	1
9	KFM Silver	2003-07-03	1
10	MST Bronze	2008-06-09	4

```sql
SELECT *
FROM vessels
WHERE owned_since LIKE '____-07%' OR owned_since LIKE '____-06%'
ORDER BY owned_since;
```

The fictional vessels must go through service every year; the above four ships need to be serviced in either June or July.

Find the estimated emission from each vessel.

vessel_id	type	env_name	emmision
1	KFM Gold	A	3.0
3	Zode Silver	A	3.0
4	MST Gold	A	3.0
6	LDS Gold	A	3.0
9	KFM Silver	A	3.0
2	KFM Silver	B	4.5
5	MST Bronze	B	4.5
7	KFM Silver	C	6.0
8	Zode Gold	C	6.0
10	MST Bronze	D	7.5

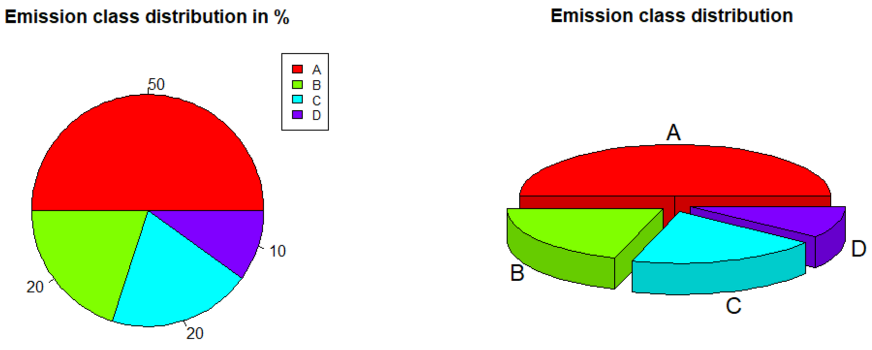


```
SELECT vessels.vessel_id, vessels.type, env_class.env_name, env_class.emmision
FROM vessels
JOIN env_class
ON vessels.env_id = env_class.env_id;
```

When choosing which vessels to send out on the long routes, it's essential to know which vessels emit the least per kilometer.

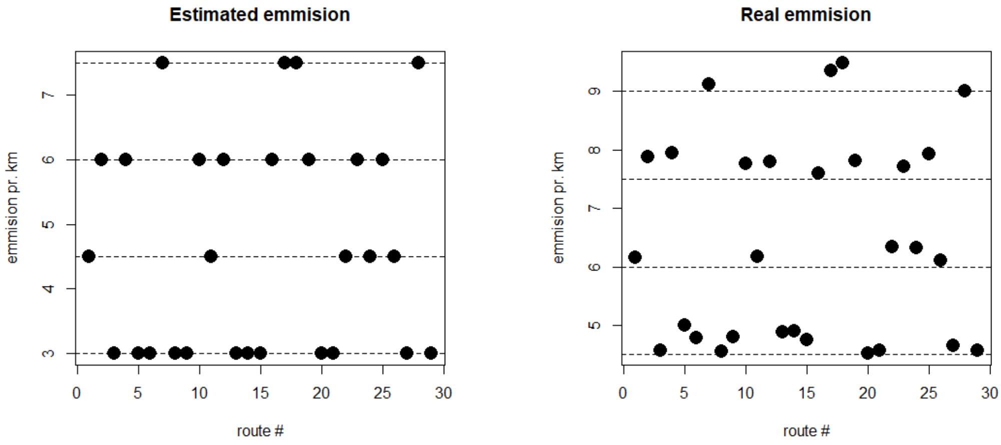
3 R

The stakeholders want to know the distribution of the environmental classes. Using R create two piecharts visualizing this distribution.
Code can found in (appx. 4.1)



Figur 2: pieCharts

There has been a problem with the latest trip. The sensor to measure emissions has broken down on the last shipment. Calculate the total emission.
The most obvious solution would be to multiply the length of the trip by the corresponding emission estimate per kilometer. But as seen in (fig. 3)(appx. 4.2), the actual emission deviates from the estimated emission, and this is due to the amount of goods a vessel carries.



Figur 3: emissionPlot

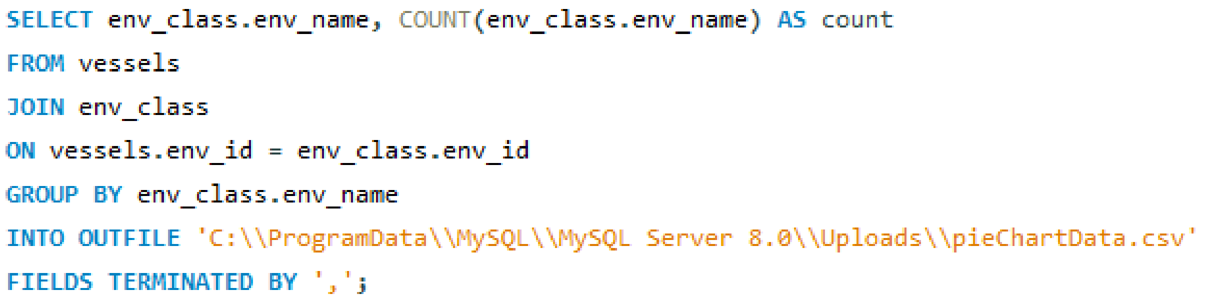
A linear regression model can predict the outcome of real emission per kilometer if given data about previous routes. Therefore let X_i^1 be amount of goods in tons, X_i^2 be estimated emission in grams per kilometer and Y_i be real emission in gramsv per kilometer. The data $(X_1^1, X_1^2, Y_1), ..., (X_n^1, X_n^2, Y_n)$ and: $X_{new}^1 = 164$, $X_{new}^2 = 3$ gets loaded into R.

Using R, the predicted emission per kilometer is approximately 4.68 gram per kilometer. (appx. 4.3)
The last route had a length of 6582 kilometers.
The latest route polluted $6582 \cdot Y_{new} = 30776.2$ grams in total.

4 Appendix

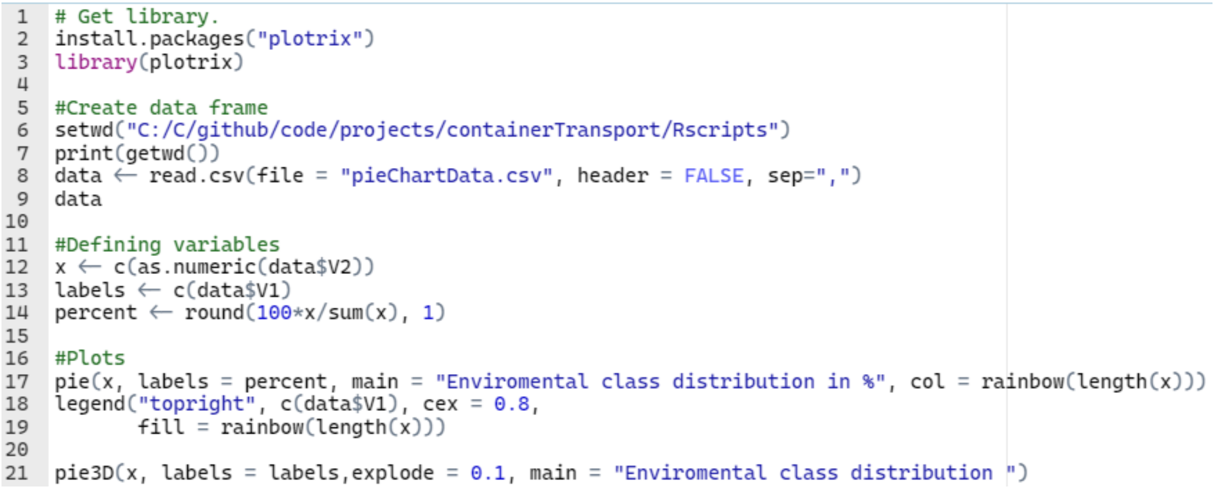
4.1 Code: Piecharts

SQL:



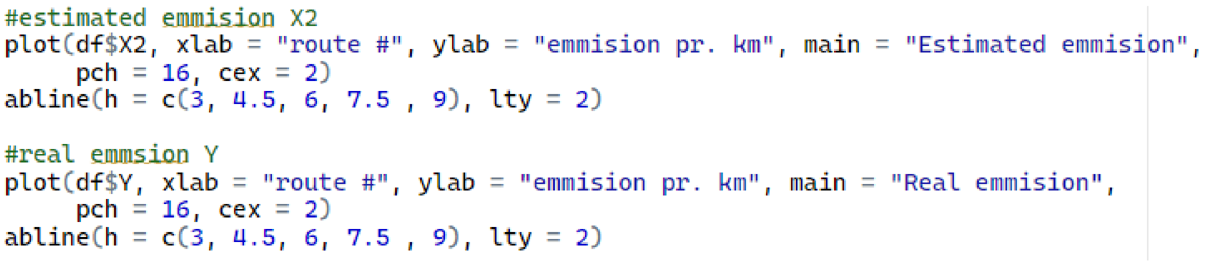
```
SELECT env_class.env_name, COUNT(env_class.env_name) AS count
FROM vessels
JOIN env_class
ON vessels.env_id = env_class.env_id
GROUP BY env_class.env_name
INTO OUTFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\pieChartData.csv'
FIELDS TERMINATED BY ',';
```

R:



```
# Get library.
install.packages("plotrix")
library(plotrix)

#Create data frame
setwd("C:/C/github/code/projects/containerTransport/Rscripts")
print(getwd())
data <- read.csv(file = "pieChartData.csv", header = FALSE, sep=",")
data

#Defining variables
x <- c(as.numeric(data$V2))
labels <- c(data$V1)
percent <- round(100*x/sum(x), 1)

#Plots
pie(x, labels = percent, main = "Enviromental class distribution in %", col = rainbow(length(x)))
legend("topright", c(data$V1), cex = 0.8,
       fill = rainbow(length(x)))

pie3D(x, labels = labels,explode = 0.1, main = "Enviromental class distribution ")
```

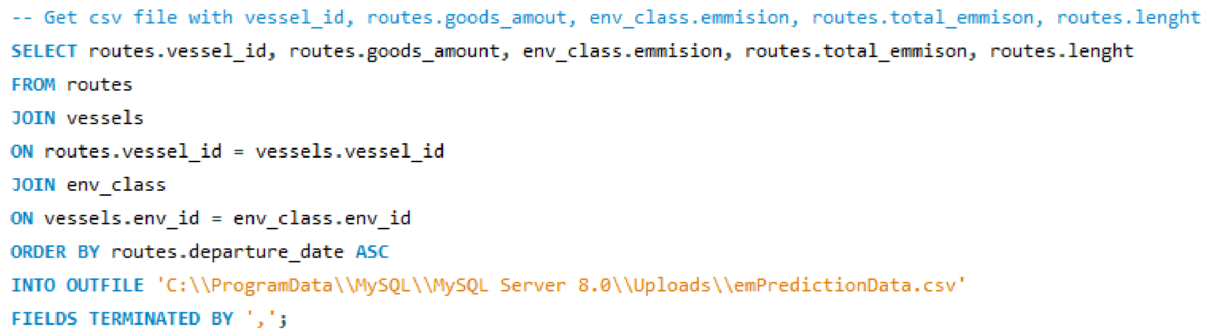
4.2 Code: Rplot emission



```
#estimated emmision X2
plot(df$X2, xlab = "route #", ylab = "emmision pr. km", main = "Estimated emmision",
     pch = 16, cex = 2)
abline(h = c(3, 4.5, 6, 7.5 , 9), lty = 2)

#real emmsion Y
plot(df$Y, xlab = "route #", ylab = "emmision pr. km", main = "Real emmision",
     pch = 16, cex = 2)
abline(h = c(3, 4.5, 6, 7.5 , 9), lty = 2)
```

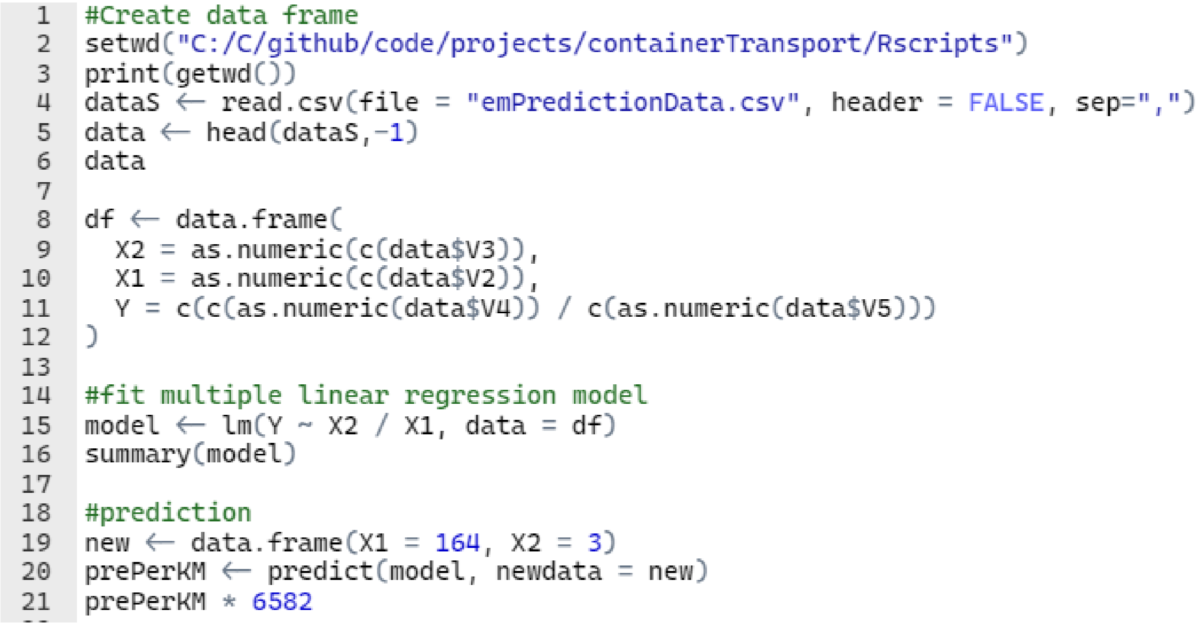
4.3 Code: Emission prediction

SQL:



```sql
-- Get csv file with vessel_id, routes.goods_amout, env_class.emmision, routes.total_emmison, routes.lenght
SELECT routes.vessel_id, routes.goods_amount, env_class.emmision, routes.total_emmison, routes.lenght
FROM routes
JOIN vessels
ON routes.vessel_id = vessels.vessel_id
JOIN env_class
ON vessels.env_id = env_class.env_id
ORDER BY routes.departure_date ASC
INTO OUTFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\emPredictionData.csv'
FIELDS TERMINATED BY ',';
```

R:



```r
#Create data frame
setwd("C:/C/github/code/projects/containerTransport/Rscripts")
print(getwd())
dataS <- read.csv(file = "emPredictionData.csv", header = FALSE, sep=",")
data <- head(dataS,-1)
data

df <- data.frame(
  X2 = as.numeric(c(data$V3)),
  X1 = as.numeric(c(data$V2)),
  Y = c(c(as.numeric(data$V4)) / c(as.numeric(data$V5)))
)

#fit multiple linear regression model
model <- lm(Y ~ X2 / X1, data = df)
summary(model)

#prediction
new <- data.frame(X1 = 164, X2 = 3)
prePerKM <- predict(model, newdata = new)
prePerKM * 6582
```