

PREVISÃO DE FOGO POSTO EM PORTUGAL 2014-2015

Data Mining I - Trabalho Prático

01 de janeiro de 2023

Trabalho Realizado por:

Joana Pereira (201805191)

Pedro Azevedo (201905966)

Pedro Santos(201904529)

Docente:

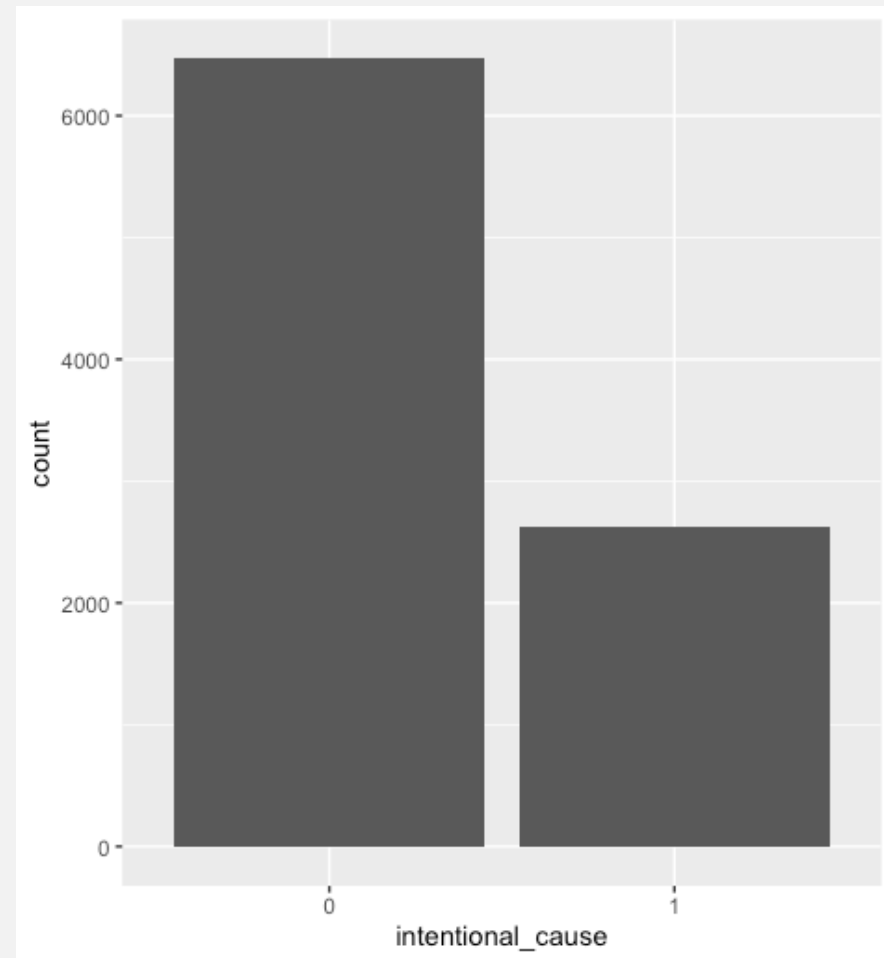
Rita Ribeiro

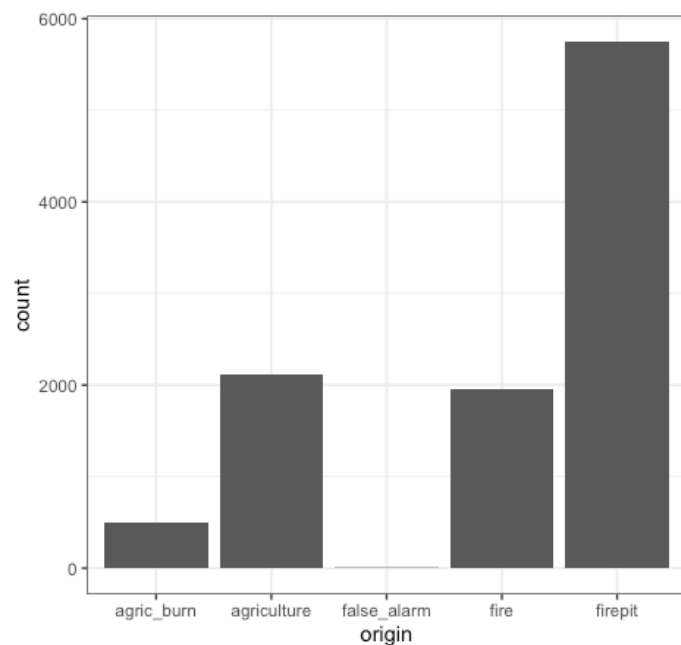
ÍNDICE

- Definição do problema
- Data Understanding
- Data Preparation
- Melhorar o conjunto de dados
- Avaliar features
- Predictive Modelling
- Conclusão

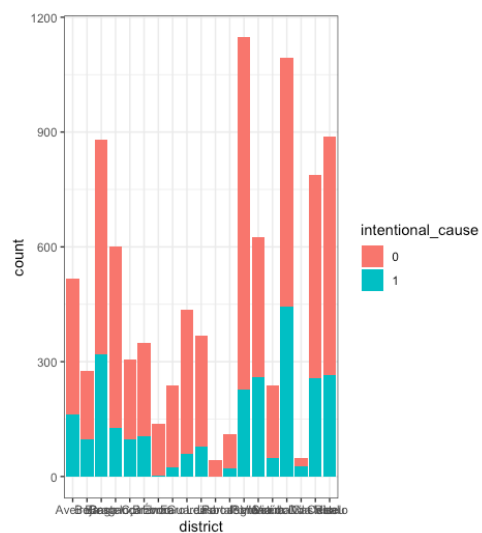
DATA UNDERSTANDING

- Características:
 - Id, region, district, municipality, parish, lat, lon, origin, alert_date, alert_hour, extinction_date, extinction_hour, firstInterv_date, firstInterv_hour, alert_source, village_área, vegetation_área, farming_área, village_veget_área, total_area
- Output
 - intentional_cause.
 - 0: 72%
 - 1: 28%





Pode, ainda, concluir que a grande parte dos fogos são iniciados por fogueira.



Viana do Castelo apresenta a maior taxa de fogo posto.

DATA
UNDERSTANDING

DATA PREPARATION

- Apagar colunas que não tem valores como alert_source e uma coluna que é muito específica e que não traria valor.

```
fire_Train_Data <- fire_Train_Data %>% select(-c(alert_source, parish))
```

- Descobrir onde existem valores nulos:

```
apply(X = is.na(fire_Train_Data), MARGIN = 2, FUN = sum)
```

- Coluna region tem poucos e por isso foram preenchidos:

```
fire_Train_Data <- fire_Train_Data %>% mutate(region = ifelse(is.na(region),  
"Ribatejo e Oeste", region))
```

DATA PREPARATION

- Por fim, apagar linhas com algum valor nulo.

```
y = c("extinction_hour", "firstInterv_date", "firstInterv_hour")
```

```
vars <- "y"
```

```
fire_Train_Data <- drop_na(fire_Train_Data, any_of(y))
```

- No caso de teste, os valores foram preenchidos:

```
fire_Test_Data <- fire_Test_Data %>% fill(extinction_date)
```

```
fire_Test_Data <- fire_Test_Data %>% fill(extinction_hour)
```

```
fire_Test_Data <- fire_Test_Data %>% fill(firstInterv_date)
```

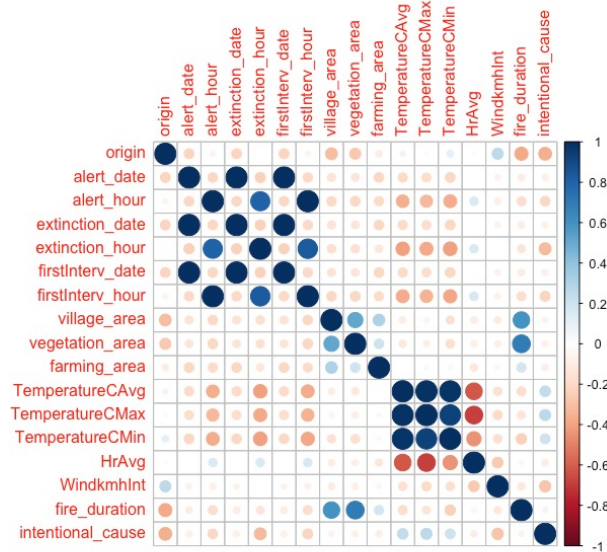
```
fire_Test_Data <- fire_Test_Data %>% fill(firstInterv_hour)
```

MELHORAR O CONJUNTO DE DADOS

- Usando uma biblioteca externa obteve-se mais características, usando a lat, lon e date do conjunto original de dados adicionou-se mais características, tanto ao conjunto de treino como de teste.

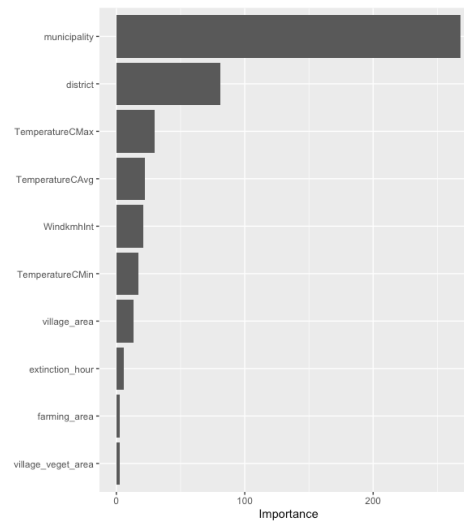
```
install_github("bczernecki/climate", force = TRUE)
```

- Assim, o conjunto final de dados ficou com as colunas:
 - District, municipality, origin, alert_date, alert_hour, extinction_date, extinction_hour, firstInterv_date, firstInterv_hour, village_area, vegetation_area, farming_area, village_veget_area, total_area, TemperatureCAvg, TemperatureCMax, TemperatureCMin, HrAvg, WindkmhInt, fire_duration e intentional_cause
- Também houve tratamento de dados.



Foi possível observar as correlações entre as features.

As caraterísticas com mais importância.



AVALIAR
FEATURES

PREDICTIVE MODELLING

- Usando as colunas que foram avaliadas como mais importantes:

```
lm_fit2 <- model_lm %>%
```

```
  fit(intentional_cause ~ district + TemperatureCMax +  
    WindkmhInt + TemperatureCAvg + TemperatureCMin +  
    village_area + extinction_hour + farming_area +  
    village_veget_area, data = fire_train)
```

- Avaliar Root Mean Squared Error, R-square e Mean absolute Error.

```
# A tibble: 3 × 3  
  .metric .estimator .estimate  
  <chr>    <chr>         <dbl>  
1 rmse     standard      0.433  
2 rsq      standard      0.0893  
3 mae      standard      0.376  
> |
```

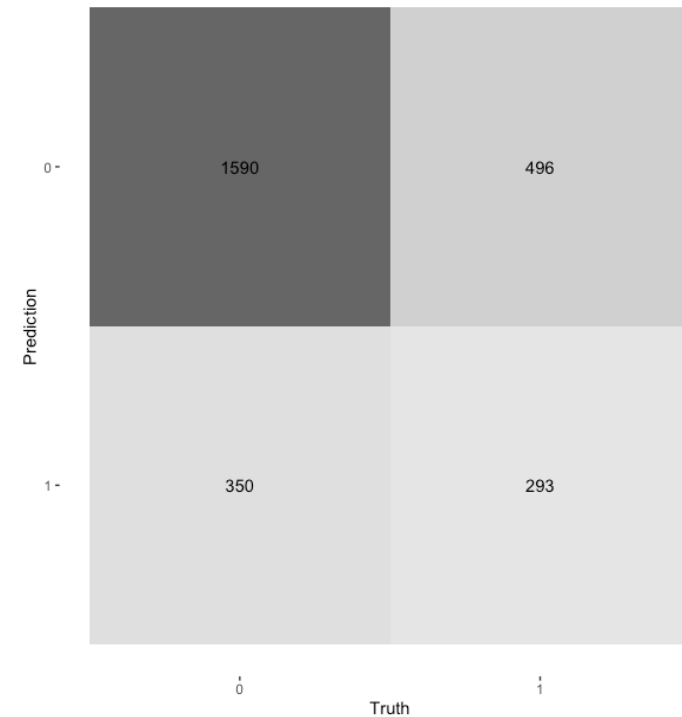
PREDICTIVE MODELLING

K Nearest-Neighbor Model

- usou-se `recipe()` como pré-processamento e sem `recipe()`, ambos com a mesma matriz de confusão, e, 69% de acurácia.

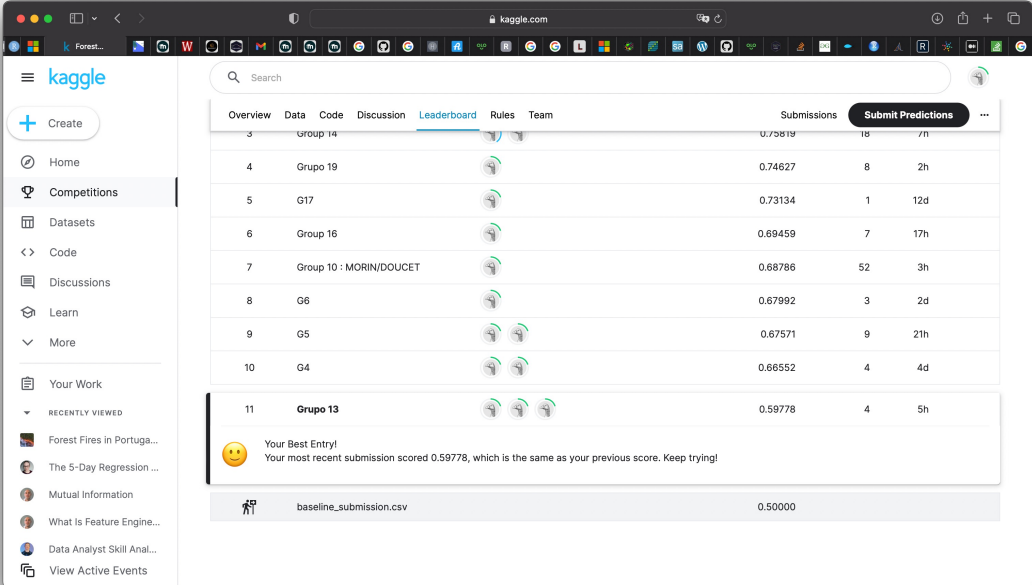
```
model_knn<-  
nearest_neighbor(mode="classification")
```

```
knn_fit <- model_knn %>%  
  fit(intentional_cause ~ district +  
    TemperatureCMax + WindkmhInt +  
    TemperatureCAvg + TemperatureCMin +  
    village_area + extinction_hour + farming_area +  
    village_veget_area, data = f_train, na.action =  
    na.exclude)
```



PREDICTIVE MODELLING

- `path <- paste(getwd(),
"/Rdata/Test_Data_noNa.rds",sep = "")`
- `fire_Test_Data <- readRDS(path)`
- `prev <- predict(knn_fit, fire_Test_Data)`
- `names(prev)[length(names(prev))] <-
"intentional_cause"`
- `prev$id <- fire_Test_Data$id`
- `prev <- prev[c("id", "intentional_cause")]`
- `write.csv(prev, "grupo13_DMI.csv",
row.names=FALSE)`



The screenshot shows the Kaggle website interface for a competition. The 'Leaderboard' tab is selected, displaying a table of participants and their scores. The table has columns for rank, group name, score, and time. A message box indicates that the user's best entry (score 0.59778) is tied with their previous submission. Below the message, a submission named 'baseline_submission.csv' with a score of 0.50000 is shown.

Overview	Data	Code	Discussion	Leaderboard	Rules	Team	Submissions	Submit Predictions	...
3	Group 14						0.75819	18	7h
4	Grupo 19						0.74627	8	2h
5	G17						0.73134	1	12d
6	Group 16						0.69459	7	17h
7	Group 10 : MORIN/DOUCET						0.68786	52	3h
8	G6						0.67992	3	2d
9	G5						0.67571	9	21h
10	G4						0.66552	4	4d
11	Grupo 13						0.59778	4	5h

Your Best Entry!
Your most recent submission scored 0.59778, which is the same as your previous score. Keep trying!

Avatar	Submission Name	Score
	baseline_submission.csv	0.50000

CONCLUSÃO

- Para ter um melhor resultado, seria melhor ter uma outra abordagem aos dados, e, ao seu processamento.
- Ainda que com algumas dificuldades, foi possível prever resultados e publicá-los.
- De salientar, que foi possível aplicar alguns modelos de previsão e, contudo, é de salientar a dificuldade em obter bons modelos de previsão.