# How to Kaggle

Carlos Sevilla Barceló

@carlosevi94

Rodrigo Gómez Rodríguez

@rodgomrod

# Summary

- What is Kaggle?
- Our experience
- Competition details

# What is Kaggle?

# Some Stats

## Users

110,624 registered users

10,000 active users per day

## Competitions

332 total competitions

72% with money prizes

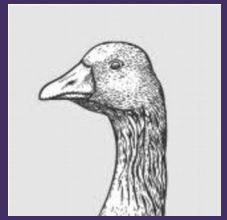
10,469,727$ given prizes

# Kaggle Tiers

| 141 Grandmasters | 1,185 Masters | 4,240 Experts | 47,772 Contributors | 57,286 Novices |
|---|---|---|---|---|







# What is Kaggle?

**José A. Guerrero**

Sevilla, Spain
Joined 8 years ago · last seen a month ago

Followers 126

Competitions Grandmaster

**Javier Tejedor Aguilera**

Non Technical Losses Detection at Endesa (Enel Group)
Seville, Spain
Joined 4 years ago · last seen in the past day

Followers 35
Following 18

Competitions Master

**Juan Galán**

**David Solis**

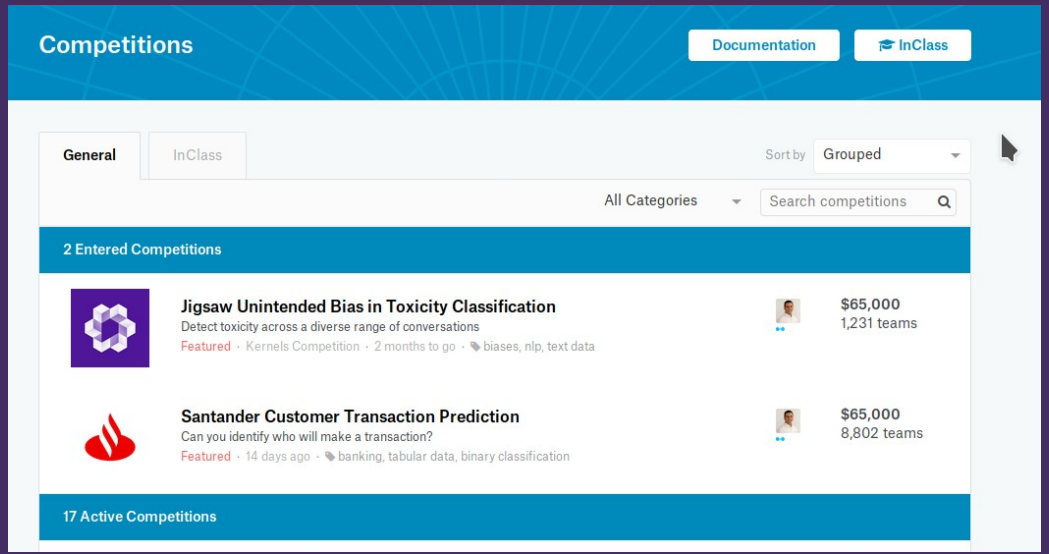
Seville, Spain
Joined 4 years ago · last seen 4 months ago

Followers 3

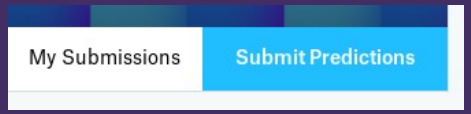
Competitions Expert

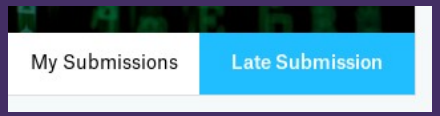
# What is Kaggle?

# Competitions



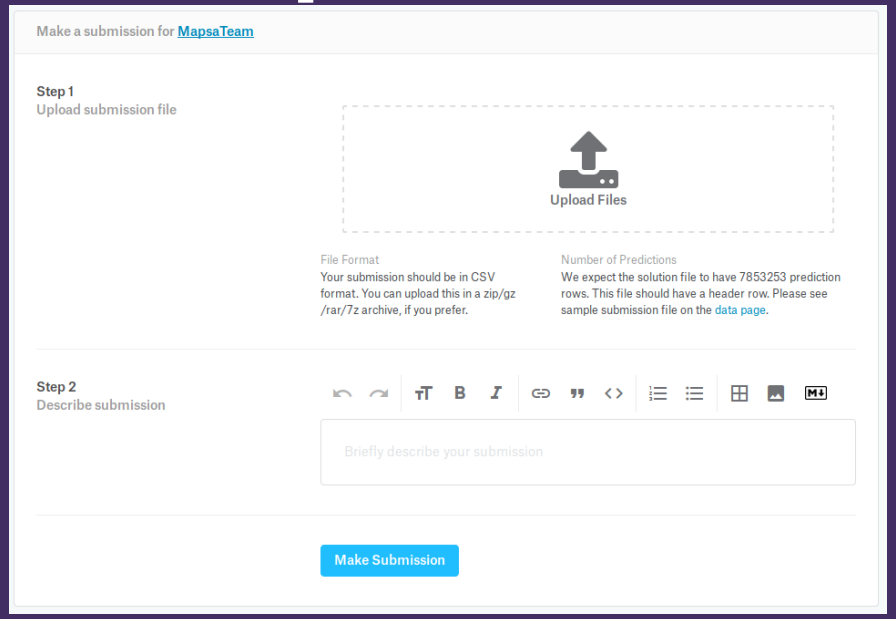
# How to compete



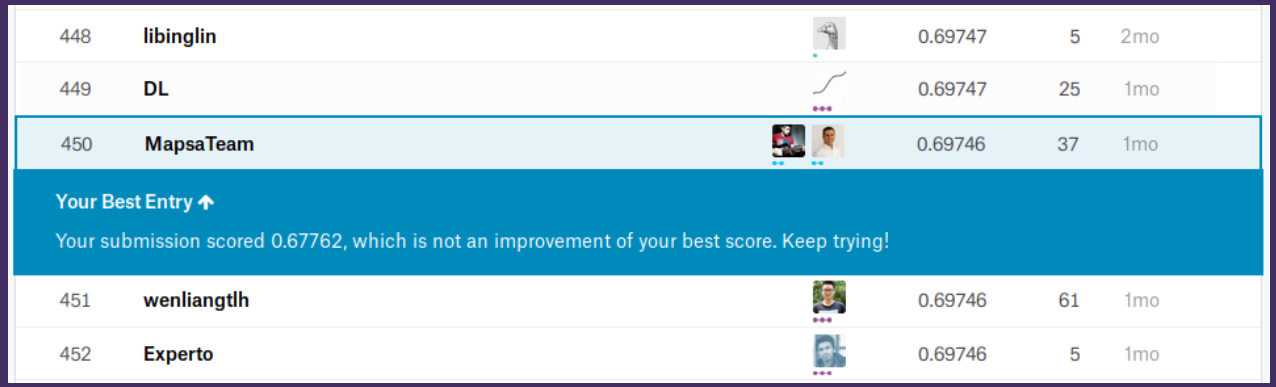



# What is Kaggle?

# How to compete



# How to compete



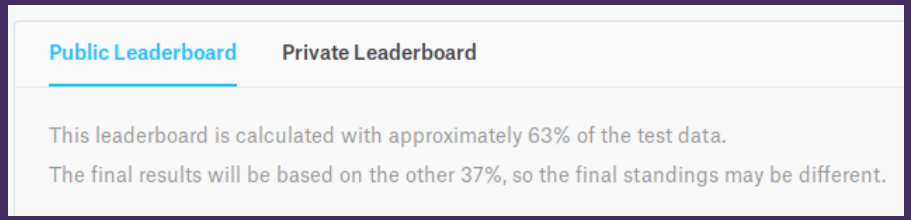
| | | | | | |
|---|---|---|---|---|---|
| 448 | libinglin | | 0.69747 | 5 | 2mo |
| 449 | DL | | 0.69747 | 25 | 1mo |
| 450 | MapsaTeam | | 0.69746 | 37 | 1mo |

Your Best Entry ↑

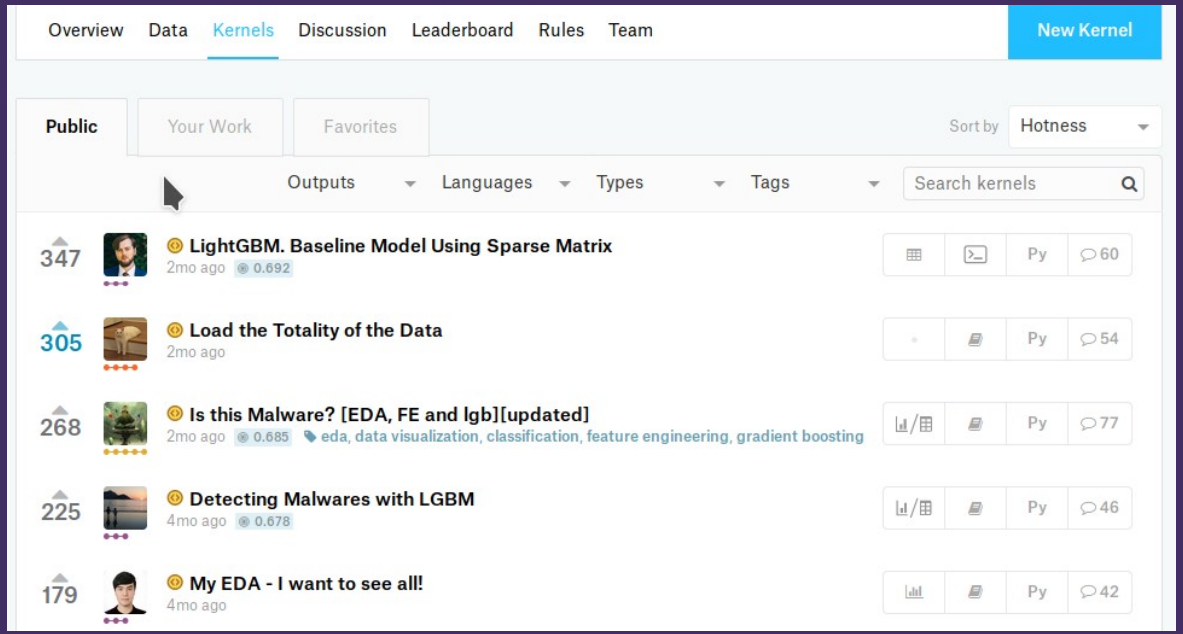
Your submission scored 0.67762, which is not an improvement of your best score. Keep trying!

| | | | | | |
|---|---|---|---|---|---|
| 451 | wenliangtlh | | 0.69746 | 61 | 1mo |
| 452 | Experto | | 0.69746 | 5 | 1mo |

# Leaderboard

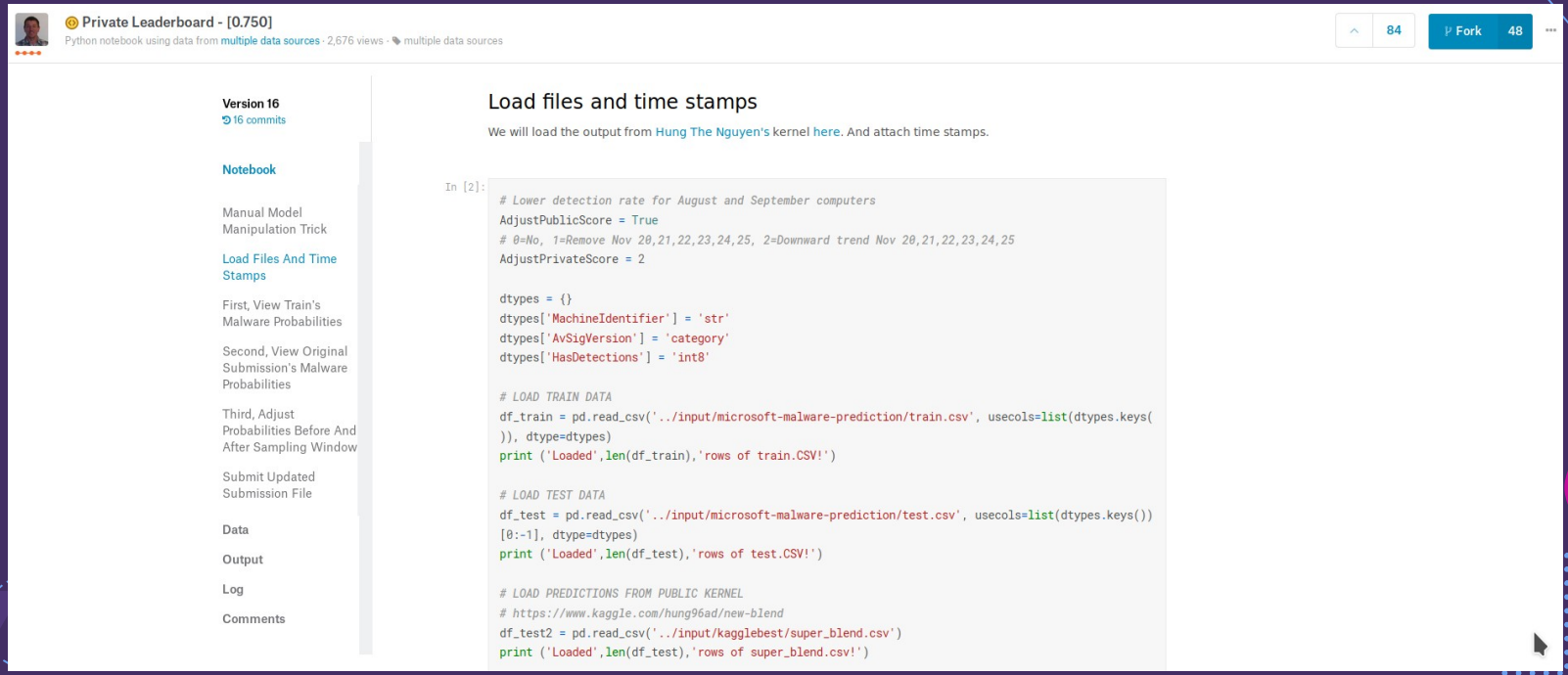


# Kernels


Overview
Data
Kernels
Discussion
Leaderboard
Rules
Team
New Kernel
Public
Your Work
Favorites
Sort by
Hotness
Outputs
Languages
Types
Tags
Search kernels
347
LightGBM. Baseline Model Using Sparse Matrix
2mo ago
0.692
Py
60
305
Load the Totality of the Data
2mo ago
Py
54
268
Is this Malware? [EDA, FE and lgb][updated]
2mo ago
0.685
eda, data visualization, classification, feature engineering, gradient boosting
Py
77
225
Detecting Malwares with LGBM
4mo ago
0.678
Py
46
179
My EDA - I want to see all!
4mo ago
Py
42

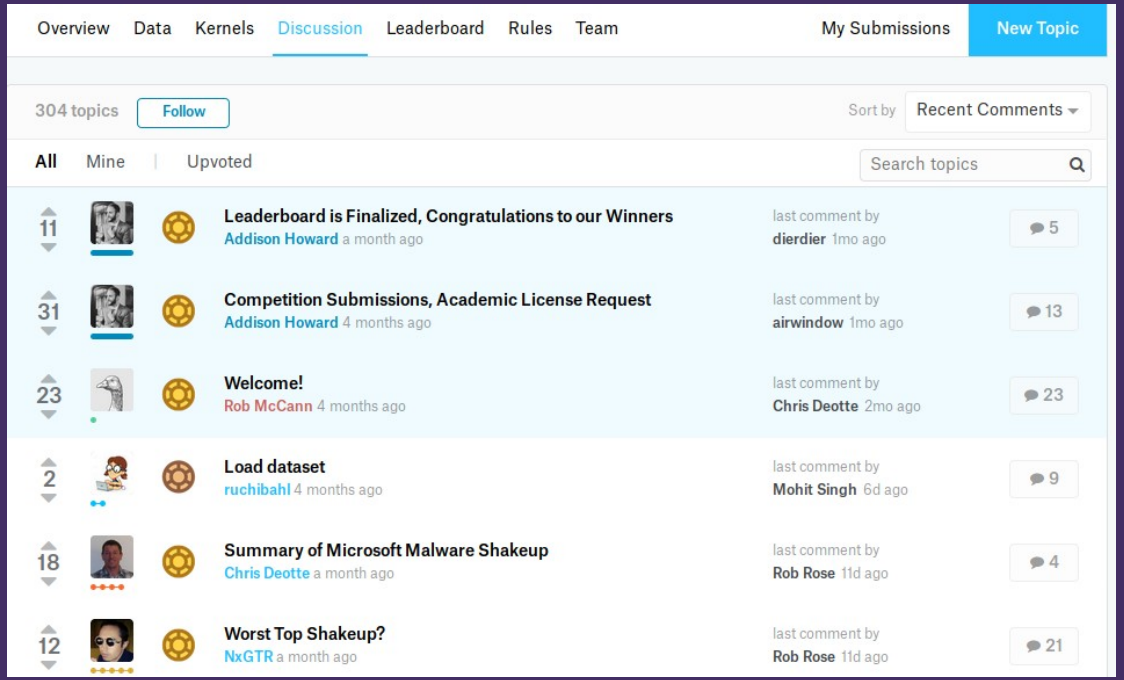
# What is Kaggle?

# Kernels



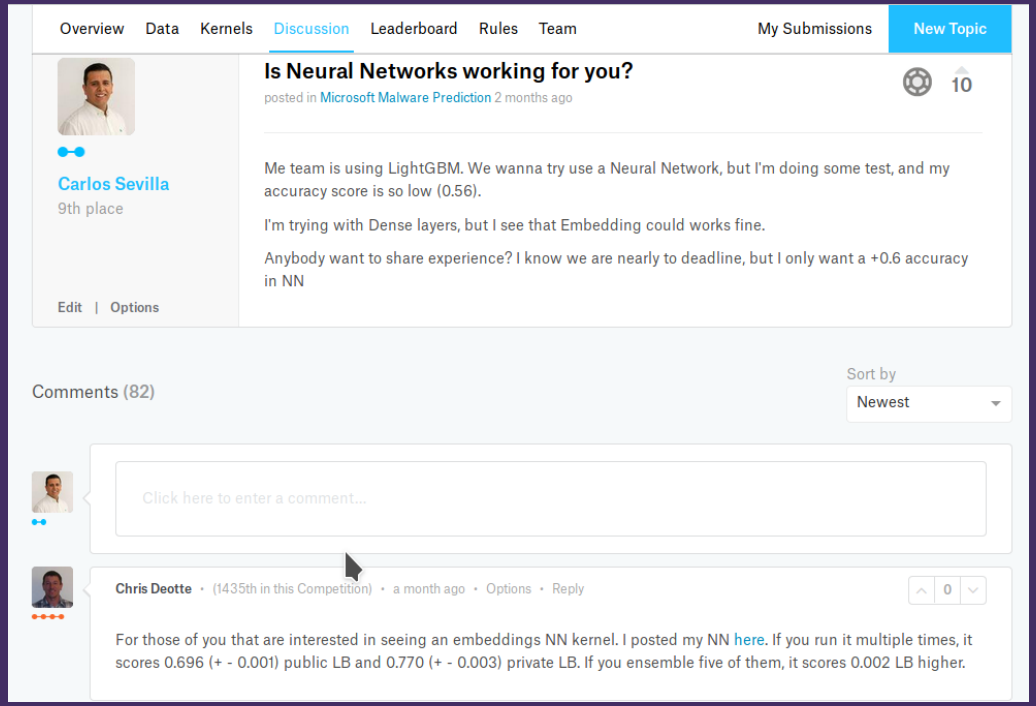
# What is Kaggle?

# Discussion

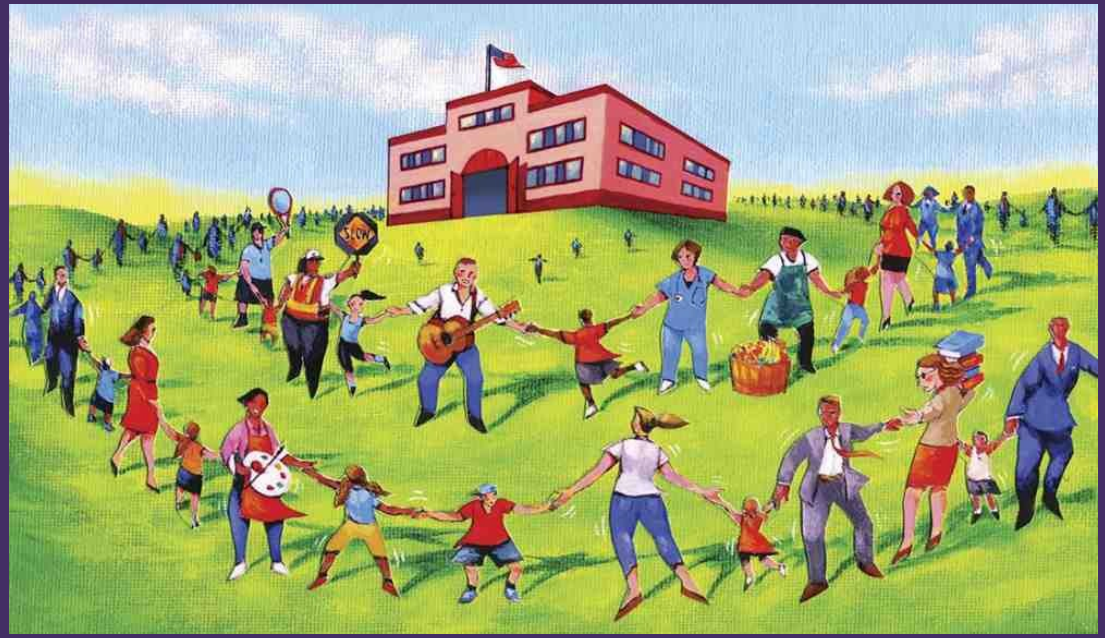


# What is Kaggle?

# Discussion

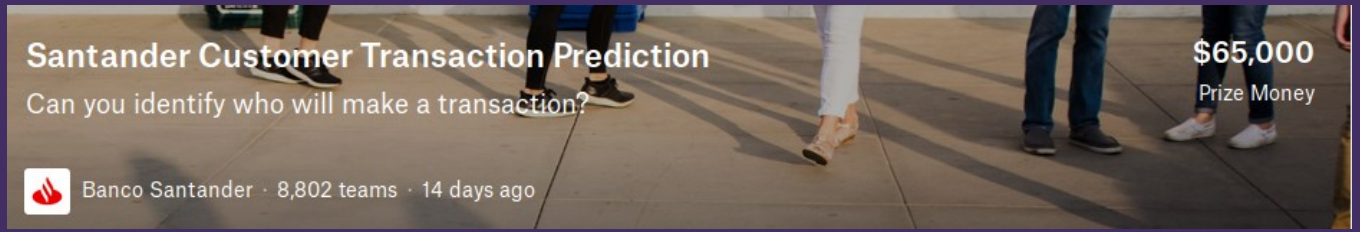


# Kaggle Community

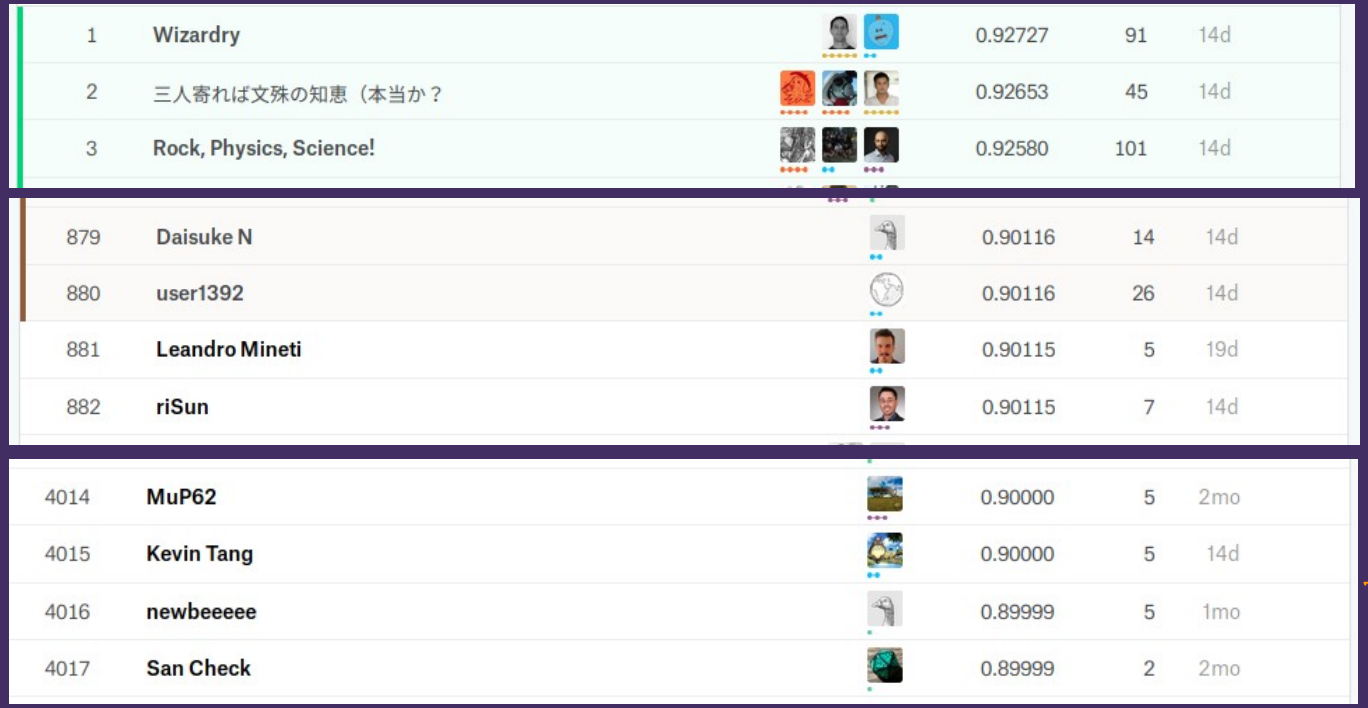


# Competitive spirit

# What is Kaggle?



# What is Kaggle?

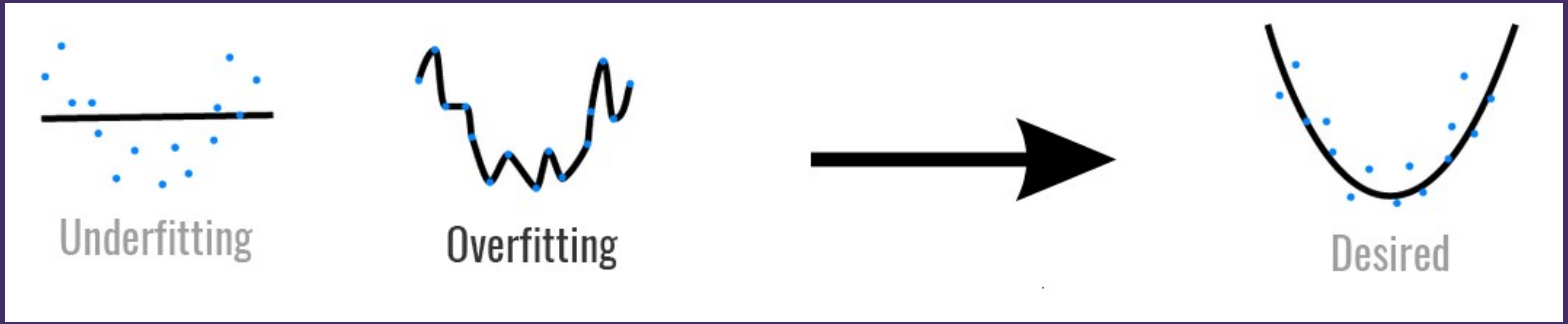


| | | | | |
|---|---|---|---|---|
| 1 | Wizardry | 0.92727 | 91 | 14d |
| 2 | 三人寄れば文殊の知恵（本当か？ | 0.92653 | 45 | 14d |
| 3 | Rock, Physics, Science! | 0.92580 | 101 | 14d |
| 879 | Daisuke N | 0.90116 | 14 | 14d |
| 880 | user1392 | 0.90116 | 26 | 14d |
| 881 | Leandro Mineti | 0.90115 | 5 | 19d |
| 882 | riSun | 0.90115 | 7 | 14d |
| 4014 | MuP62 | 0.90000 | 5 | 2mo |
| 4015 | Kevin Tang | 0.90000 | 5 | 14d |
| 4016 | newbeeeee | 0.89999 | 5 | 1mo |
| 4017 | San Check | 0.89999 | 2 | 2mo |

# Nobody loses at Kaggle

Every second that you’re testing some stuff, you’re learning. And that improves you.

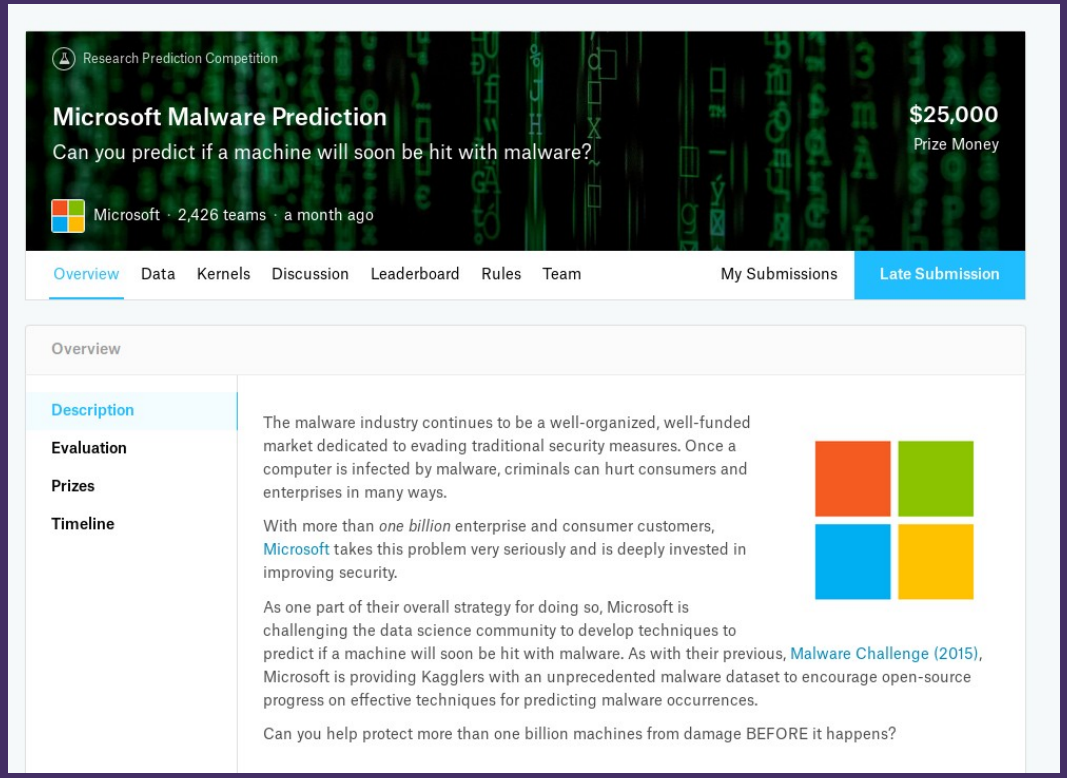
# What is Kaggle?



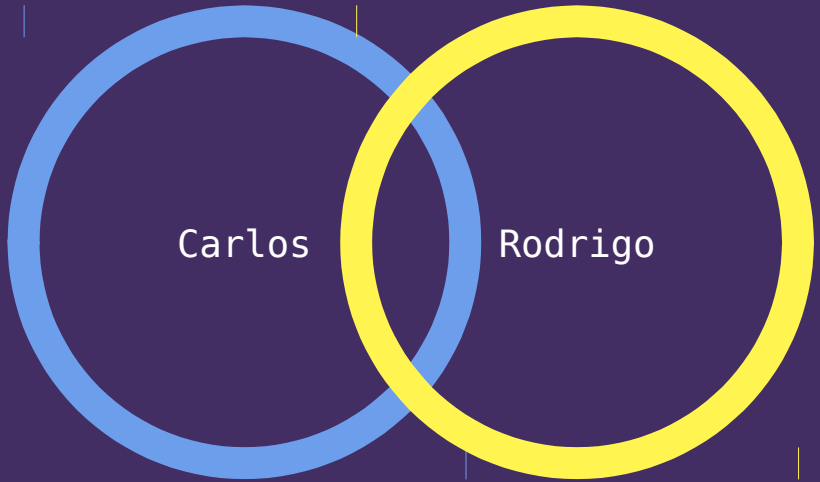
# Mapsa Team

# Our experience

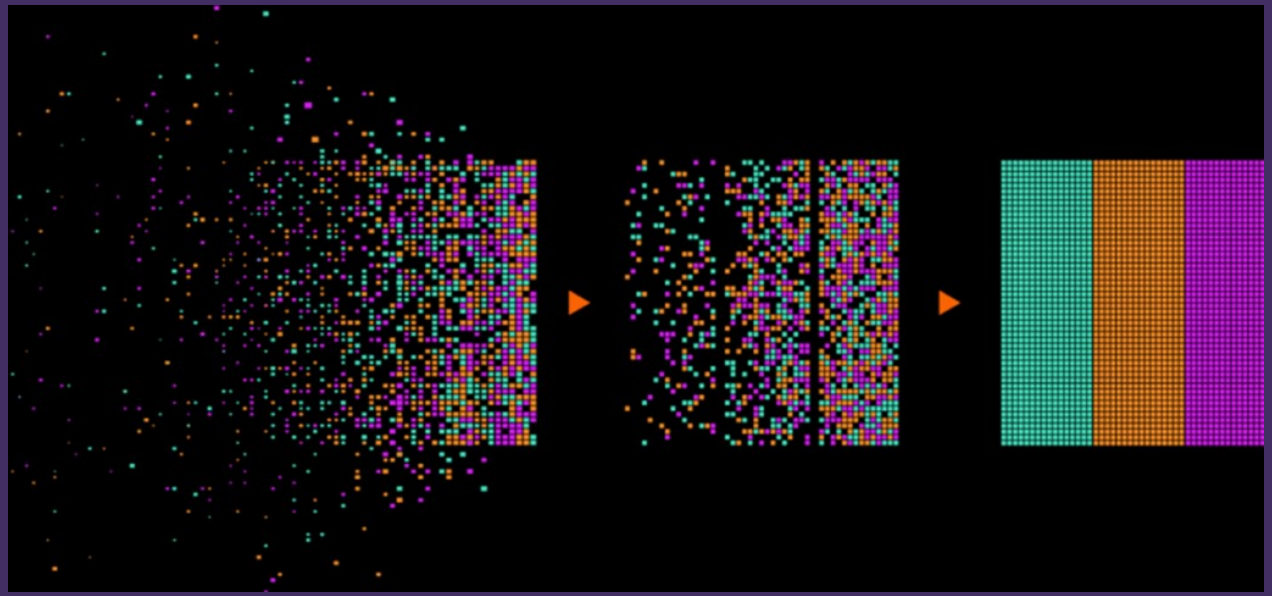

# Our experience



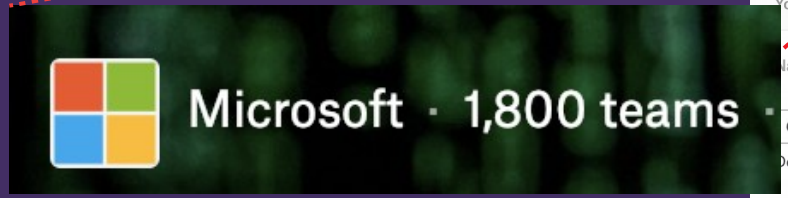
# Our experience

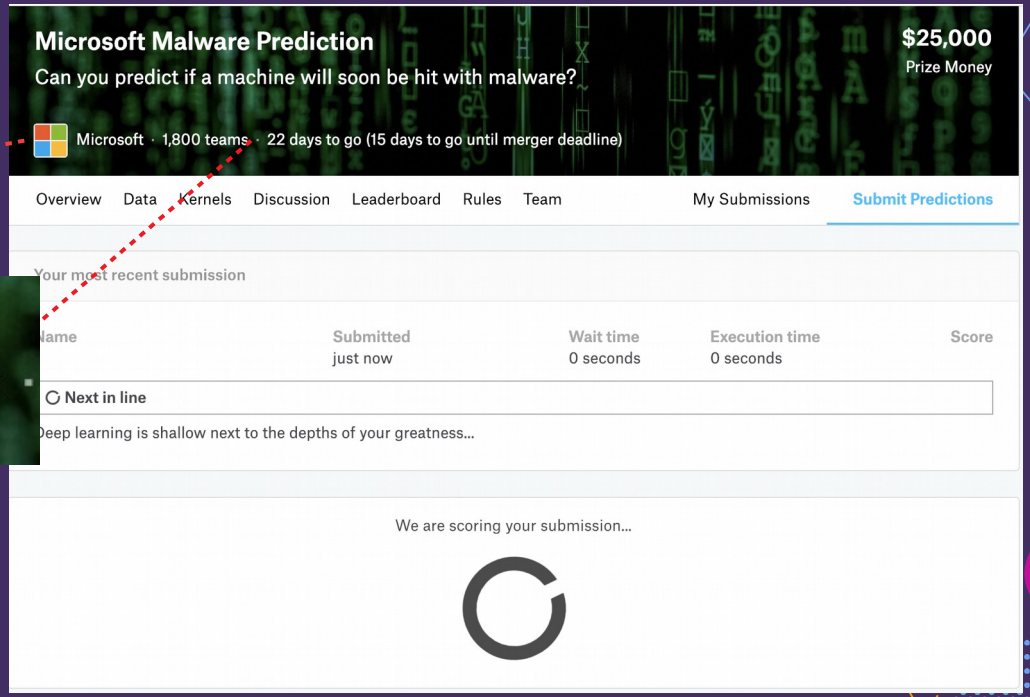


# Our experience



# Our experience



# Our experience

| | | | | | |
|---|---|---|---|---|---|
| 1801 | Badner | | 0.468 | 1 | 6h |
| 1802 | MapsaTeam | | 0.340 | 1 | 9m |

**Your Best Entry ↑**

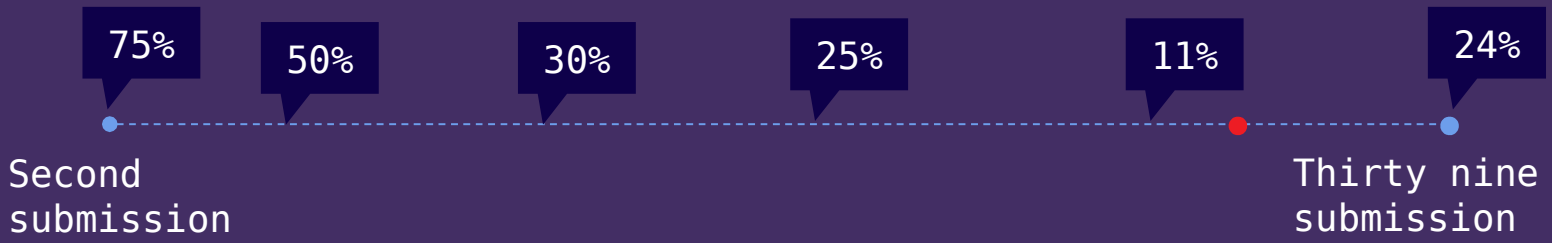
Your submission scored 0.340 Tweet this!

| | | | | | |
|---|---|---|---|---|---|
| 1803 | dict | | 0.336 | 1 | 7d |

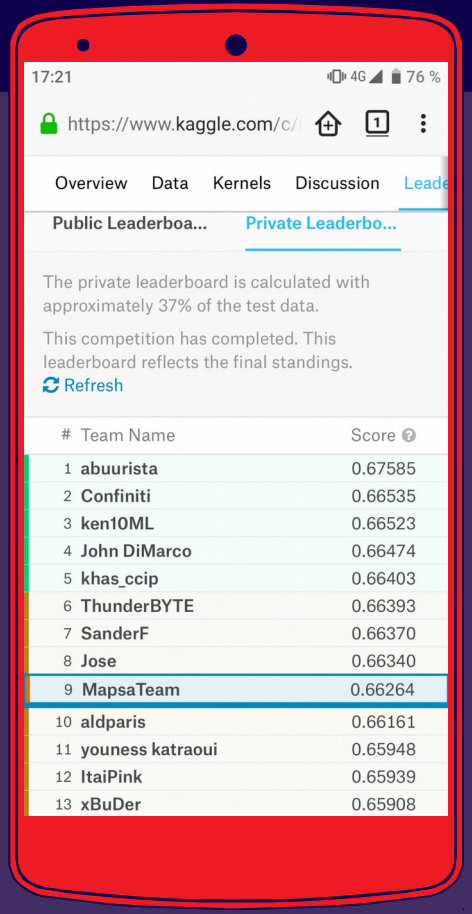
# Our experience

**Mapsa Team**

# Our experience

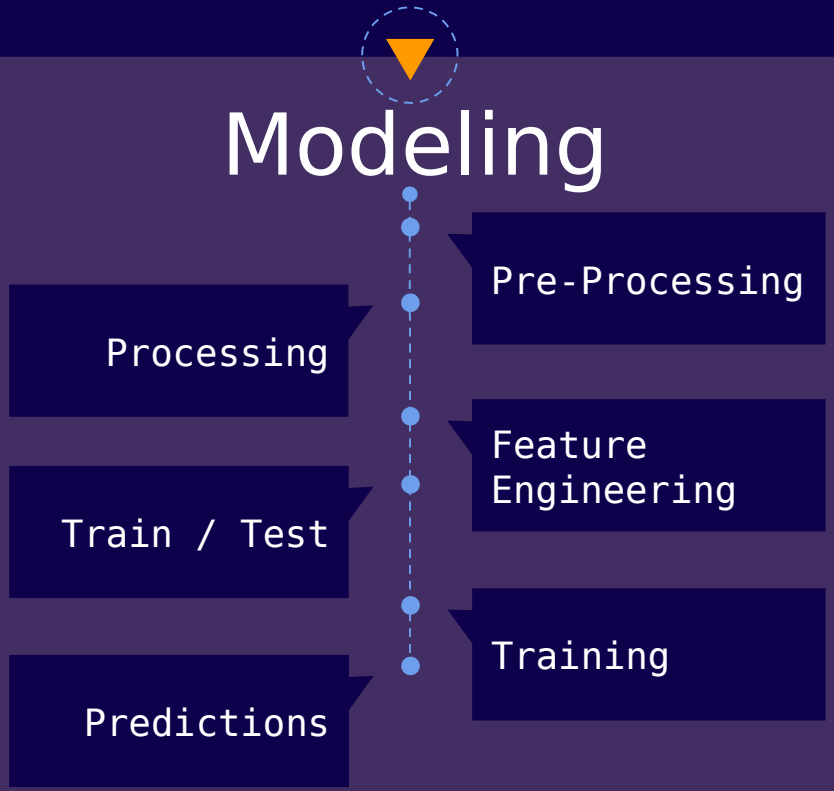


# Our experience




# Competition details

# Competition details



# Competition details

## Pre-Processing

- Train and test union
- Numerical / categorical split
- Data cleaning

```
def transformaciones_OSEdition(x):
    if x == '#':
        return None
    elif x == '00426-OEM-8992662-00006':
        return 'Ultimate'
    elif x == 'HomePremium' or x == 'HomeBasic':
        return 'Home'
    elif x == 'Window 10 Enterprise' or x == 'Enterprise 2015 LTSB' or x == 'EnterpriseN' \
         or x == 'EnterpriseG':
        return 'Enterprise'
    elif x == 'ServerDatacenterACor' or x == 'ServerDatacenterEval':
        return 'ServerDatacenter'
    elif x == 'ProfessionalSingleLanguage' or x == 'PRO' or x == 'Pro' or x == 'professional' \
         or x == 'ProfessionalCountrySpecific':
        return 'Professional'
    elif x == 'ProfessionalEducationN' or x == 'EducationN' or x == 'ProfessionalEducation':
        return 'Education'
    elif x == 'CloudN':
        return 'Cloud'
    elif x == 'ProfessionalWorkstationN':
        return 'ProfessionalWorkstation'
    else:
        return x

udf_OSEdition = udf(lambda z: transformaciones_OSEdition(z), StringType())
```

# Competition details

## Processing

- Label Encoding
- NaN imputation
- Version split & L.E.

```
data = data.withColumn('Census_InternalBatteryType_informed',
                       when(col('Census_InternalBatteryType').isNotNull(),1).otherwise(0))

print('\Pipeline de Indexers paras las columnas {0}\n'.format(cols_le))
indexers = [StringIndexer(inputCol=c, outputCol=c+"_index", handleInvalid="keep").fit(data) for c in cols_le]
pipeline = Pipeline(stages=indexers)
data = pipeline.fit(data).transform(data)
data = data.drop(*cols_le)

print("Persist intermedio 0\n")
data.persist()
print(data.first())

# Transformamos las columnas de versiones "x.y.z.t"
# Conversion a LabelEncoding

# print('Transformacion columnas de versiones\n')
print('\tCensus_OSVersion\n')
data = data.withColumn('Census_OSVersion_0', concat(split(data['Census_OSVersion'], '\.')[0],
                                                    split(data['Census_OSVersion'], '\.')[1]))\
    .withColumn('Census_OSVersion_1', concat(split(data['Census_OSVersion'], '\.')[0],
                                             split(data['Census_OSVersion'], '\.')[1],
                                             split(data['Census_OSVersion'], '\.')[2]))
```

# Competition details

## Feature Engineering

- Frequency encoding
- Dates
- Extra info.

```
N1 = ['Census_OSEdition', 'SmartScreen', 'Census_OSBranch', 'Census_OSSkuName']

N2 = ['Census_ChassisTypeName', 'Census_FlightRing', 'Census_MDC2FormFactor', 'ProductName']

N3 = ['Census_PrimaryDiskTypeName', 'Census_PowerPlatformRoleName', 'Census_ProcessorClass']

N4 = ['Processor', 'Census_OSInstallTypeName', 'OsVer', 'Census_GenuineStateName', 'PuaMode']

N5 = ['Census_OSVersion', 'EngineVersion', 'AppVersion', 'AvSigVersion']

write_path = 'data/df_groupby_cat_0'

spark = SparkSession.builder.appName("Microsoft_Kaggle").getOrCreate()

data = spark.read.csv('data/df_cat_prepro_0/*.csv', header=True, inferSchema=True).select(N1+N2+N3+N4+N5+['MachineIdentifier'])

print('Guardamos el DF en {}\n'.format(write_path))
data.join(data.groupBy(N1).agg(count('*').alias('count1')), N1, 'left')\
    .join(data.groupBy(N2).agg(count('*').alias('count2')), N2, 'left')\
    .join(data.groupBy(N3).agg(count('*').alias('count3')), N3, 'left')\
    .join(data.groupBy(N4).agg(count('*').alias('count4')), N4, 'left')\
    .join(data.groupBy(N5).agg(count('*').alias('count5')), N5, 'left')\
    .select('MachineIdentifier', 'count1', 'count2', 'count3', 'count4', 'count5')\
    .write.csv(write_path, sep=',', mode="overwrite", header=True)
```

# Competition details

## Train & Test

- train.csv generation
- test.csv generation

```
df_gby_cat = spark.read.csv('data/df_groupby_cat_0/*.csv', header=True).repartition(80, ['MachineIdentifier'])
df_gby_cat.persist()
df_gby_cat.count()

df_gby_num = spark.read.csv('data/df_groupby_num_0/*.csv', header=True).repartition(80, ['MachineIdentifier'])
df_gby_num.persist()
df_gby_num.count()


full_df = df_num.join(df_cat, ['MachineIdentifier'])\
                .join(df_dates, ['MachineIdentifier'])\
                .join(df_kmeans, ['MachineIdentifier'])\
                .join(df_avsigver, ['MachineIdentifier'])\
                .join(df_gby_cat, ['MachineIdentifier'])\
                .join(df_gby_num, ['MachineIdentifier']).repartition(80, ['MachineIdentifier'])


full_df.persist()
full_df.count()

train = full_df.filter(col('HasDetections').isNotNull()).fillna('-1')

test = full_df.filter(col('HasDetections').isNull()).fillna('-1')
```

# Competition details

## Training

- LightGBM
- XGBoost
- CatBoost
- Keras

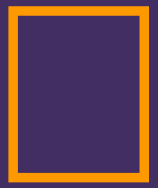
```
print('Comienza entrenamiento del modelo LightGBM')
lgb_model = lgb.LGBMClassifier(**params)

ft_importances = np.zeros(X_train.shape[1])

counter = 1
for train_index, test_index in skf.split(train_ids, y_train):
    print('Fold {}\n'.format(counter))

    X_fit, X_val = X_train.iloc[train_index, :], X_train.iloc[test_index, :]
    y_fit, y_val = y_train.iloc[train_index], y_train.iloc[test_index]

    lgb_model.fit(X_fit,
                  y_fit,
                  eval_set=[(X_val, y_val)],
                  verbose=100,
                  early_stopping_rounds=100)

    del X_fit
    del X_val
    del y_fit
    del y_val
    del train_index
    del test_index
    gc.collect()
```

# Competition details

## Predictions

- LightGBM
- Ensemble
- Submit to Kaggle

```
X_test = test.loc[:, sel_cols]
X_machines = test.loc[:, 'MachineIdentifier']
del test
del list_
gc.collect()

lgb_test_preds = np.zeros(X_test.shape[0])

for i in range(1, k+1):
    model = joblib.load('saved_models/{}_{}.pkl'.format(model_name, i))
    print('Realizando predicciones. FOLD = {}'.format(i))
    lgb_test_preds += model.predict_proba(X_test)[:, 1]

    del model
    gc.collect()

del X_test
gc.collect()

print('Haciendo la media y guardando CSV')
final_prds = lgb_test_preds/k

df_prds = pd.DataFrame({'MachineIdentifier': X_machines, 'HasDetections': final_prds})

df_prds.to_csv('submissions/{}.csv'.format(model_name), index=None)
```



# Thanks!

**Any questions?**