

Polars Descriptive Statistics Assignment

Aircraft wildlife strikes data | 1990 - 2015

In this exercise, we will extract and analyze aircraft wildlife strikes data, and we will determine the probability of each part of an aircraft getting damaged by an aircraft wildlife strike

```
In [ ]: # Import the necessary libraries
```

```
import polars as pl
import matplotlib.pyplot as plt
import requests
import io
```

```
In [ ]: # Read our data from Google Drive
```

```
file_id = "1TAD7Uyc9PjByt_q13uvGXGeubXnujnUi"
url = f"https://drive.google.com/uc?id={file_id}"

# Download the contents of the CSV file
download = requests.get(url).content

# Read the CSV file into a polars DataFrame
df = pl.read_csv(
    io.StringIO(download.decode("utf-8")), low_memory=False, infer_schema_length=10
)
```

```
In [ ]: # Explore the data
```

```
df.head()
```

Out[]: shape: (5, 66)

Record ID	Incident Year	Incident Month	Incident Day	Operator ID	Operator	Aircraft	Aircraft Type	Aircraft Make
i64	i64	i64	i64	str	str	str	str	str
127128	1990	1	1	"DAL"	"DELTA AIR LINE..."	"B-757-200"	"A"	"148"
129779	1990	1	1	"HAL"	"HAWAIIAN AIR"	"DC-9"	"A"	"583"
129780	1990	1	2	"UNK"	"UNKNOWN"	"UNKNOWN"	null	null
2258	1990	1	3	"MIL"	"MILITARY"	"A-10A"	"A"	"345"
2257	1990	1	3	"MIL"	"MILITARY"	"F-16"	"A"	"561"

In []: df.describe()


Out[]:

Propeller Strike	Propeller Damage	Wing or Rotor Strike	Wing or Rotor Damage	Fuselage Strike	Fuselage Damage	Landing Gear Strike	Landing Gear Damage	Tail Strike
f64	f64	f64	f64	f64	f64	f64	f64	f64
174104.0	174104.0	174104.0	174104.0	174104.0	174104.0	174104.0	174104.0	174104.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.020086	0.003326	0.119159	0.024009	0.102703	0.004733	0.046242	0.005813	0.011235
0.140294	0.057572	0.323976	0.153076	0.303571	0.068633	0.21001	0.076019	0.105397
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

In []: df.median()

Out[]:

ent	Incident	Incident	Operator	Operator	Aircraft	Aircraft	Aircraft	Aircraft	Aircraft	Aircraft
'ear	Month	Day	ID			Type	Make	Model	Mass	
f64	f64	f64	str	str	str	str	str	str	f64	
17.0	8.0	16.0	null	null	null	null	null	null	4.0	



Now we are going to calculate the probability of each part of the flight getting damaged and plot these probabilities

```
In [ ]: strikes = {}
for c in df.columns:
    column_name = c.split(" ")
    # print(len(col_sep), col_sep)
    if len(column_name) > 1 and column_name[1] == "Strike":
        strikes[column_name[0]] = df[column_name[0] + " Damage"].sum() / df[c].sum()
```

```
In [ ]: # Calculate the probability of each part of the aircraft getting damaged and find t
plt.bar(strikes.keys(), strikes.values())
plt.xticks(rotation=90)
plt.title("Aircraft Part Damage Probability")
print(max(strikes, key=strikes.get))
```

Lights

Aircraft Part Damage Probability

