Machine Learning (CCS4340)

Lab 01 : Download, prepare and save the Credit Approval Dataset

In this notebook, you will find guidelines to download, prepare, and store the Credit Approval Dataset from the UCI Machine Learning Repository.

Download the data

Follow these guidelines to download the data:

- Visit the UCI website (http://archive.ics.uci.edu/ml/machine-learning-databases/credit-screening/)
- Click on crx.data to download the data.
- Save crx.data in the same folder that contains this notebook.
- You can find more information about this particular dataset here. https://archive.ics.uci.edu/ml/datasets/credit+approval

In [1]:

```
import random
import numpy as np
import pandas as pd
```

In [42]:

```
# load data
data = pd.read_csv("crx.data", header=None)
#data.head()

# Create variable names according to UCI Machine Learning
# Repository's information:
varnames = [f"A{s}" for s in range(1, 17)]
#print(varnames)

# Add column names to dataset
data.columns = varnames

# Replace ? by np.nan
data = data.replace("?", np.nan)

# Cast variables to correct datatypes:
data["A2"] = data["A2"].astype("float")
data["A14"] = data["A14"].astype("float")

# Encode target to binary notation:
data["A16"] = data["A16"].map({"+":1, "-":0})

# Rename target:
data.rename(columns={"A16": "target"}, inplace=True)

# Display first 5 rows of data:
data.head(15)
```

Out[42]:

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	target
0	b	30.83	0.000	u	g	w	v	1.250	t	t	1	f	g	202.0	0	1
1	a	58.67	4.460	u	g	q	h	3.040	t	t	6	f	g	43.0	560	1
2	a	24.50	0.500	u	g	q	h	1.500	t	f	0	f	g	280.0	824	1
3	b	27.83	1.540	u	g	w	v	3.750	t	t	5	t	g	100.0	3	1
4	b	20.17	5.625	u	g	w	v	1.710	t	f	0	f	s	120.0	0	1
5	b	32.08	4.000	u	g	m	v	2.500	t	f	0	t	g	360.0	0	1
6	b	33.17	1.040	u	g	r	h	6.500	t	f	0	t	g	164.0	31285	1
7	a	22.92	11.585	u	g	cc	v	0.040	t	f	0	f	g	80.0	1349	1
8	b	54.42	0.500	y	p	k	h	3.960	t	f	0	f	g	180.0	314	1
9	b	42.50	4.915	y	p	w	v	3.165	t	f	0	t	g	52.0	1442	1
10	b	22.08	0.830	u	g	c	h	2.165	f	f	0	t	g	128.0	0	1
11	b	29.92	1.835	u	g	c	h	4.335	t	f	0	f	g	260.0	200	1
12	a	38.25	6.000	u	g	k	v	1.000	t	f	0	t	g	0.0	0	1
13	b	48.08	6.040	u	g	k	v	0.040	f	f	0	f	g	0.0	2690	1
14	a	45.83	10.500	u	g	q	v	5.000	t	t	7	t	g	0.0	0	1

In [43]:

```
# Add missing values at random positions.

# Set seed for reproducibility:
random.seed(9001)

# get the random position indexes:
values = list(set([random.randint(0, len(data)) for p in range(0, 100)]))
print(values)

# Add mising data:
data.loc[values, ["A3","A8","A9","A10"]] = np.nan

# Check propotion of missing data:
data.isnull().sum()
```

```
[512, 2, 5, 523, 12, 525, 526, 528, 532, 536, 539, 27, 543, 37, 551, 552, 44, 564,
55, 60, 577, 69, 582, 583, 76, 80, 602, 607, 101, 620, 623, 113, 117, 630, 119, 12
7, 128, 647, 649, 650, 142, 659, 675, 676, 167, 176, 187, 196, 221, 225, 226, 228,
238, 243, 253, 256, 259, 262, 275, 284, 285, 294, 296, 298, 299, 308, 309, 312, 31
3, 315, 331, 357, 362, 363, 369, 377, 384, 387, 401, 405, 422, 430, 434, 436, 438,
442, 450, 454, 466, 497, 503, 507]
```

```
Out[43]: A1        12
         A2        12
         A3        92
         A4         6
         A5         6
         A6         9
         A7         9
         A8        92
         A9        92
         A10       92
         A11        0
         A12        0
         A13        0
         A14       13
         A15        0
         target     0
         dtype: int64
```

```
In [45]: # Save Dataset
         data.to_csv("credit_approval_uci.csv", index=False)
```