In [ ]:

```
from google.colab import drive
!git clone https://github.com/hsmmi/Exact-Acceleration-of-K-Means-and-K-Means-pa
drive.mount('/content/drive')
!unzip /content/drive/MyDrive/Public/dataset.zip -d /content/Exact-Acceleration-
drive.flush_and_unmount()
```

```
Mounted at /content/drive
```

In [ ]:

```
import sys
sys.path.append('/content/Exact-Acceleration-of-K-Means-and-K-Means-parallel')
!pip install pickledb
!pip install vptree
```

In [53]:

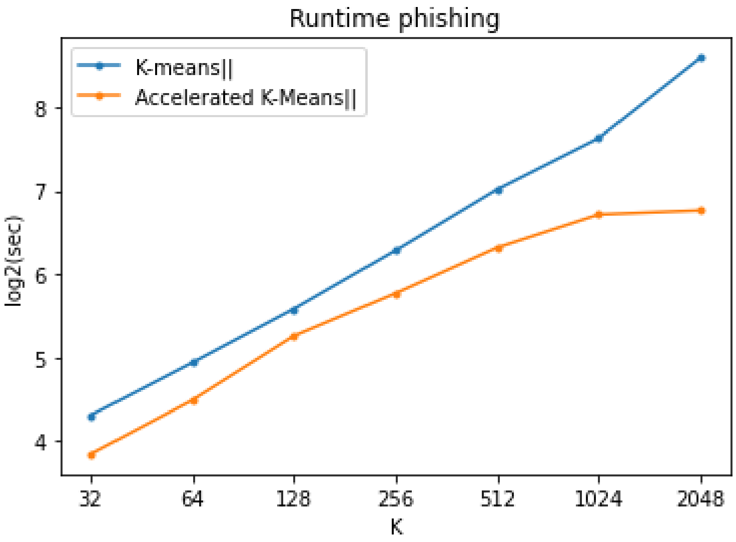
```
from time import time
from matplotlib import pyplot as plt
import numpy as np
import pandas as pd
from akll import AKLL
from dataset import Dataset
from kll import KLL
import warnings

warnings.filterwarnings("ignore")

root_dir = "/content/Exact-Acceleration-of-K-Means-and-K-Means-parallel"
dataset_name = "phishing"

dataset = Dataset(f"{root_dir}/dataset/{dataset_name}.csv")

start_range, end_range = 5, 12
k = np.arange(start_range, end_range)

uid = np.random.randint(0, 100)

kll_log = []
akll_log = []

for i in k:
    K = 2**i

    print(f"running kll for k: {K}")
    kll = KLL(dataset)
    ts = time()
    centers = kll.fit(K)
    te = time()
    kll_log.append(te - ts)
    print(f"kll runs in: {te - ts}sec")

    print(f"running akll for k: {K}")
    akll = AKLL(dataset)
    ts = time()
    centers = akll.fit(K)
    te = time()
    akll_log.append(te - ts)
    print(f"akll runs in: {te - ts}sec")
```

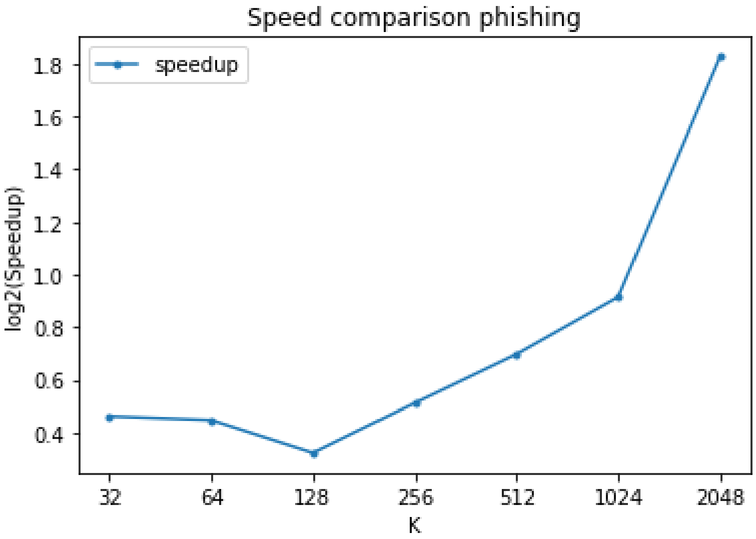
```
x_axis = (2**k).astype(str)

log_kll_log = np.log(kll_log)
log_akll_log = np.log(akll_log)
plt.figure(facecolor="white", figsize=(6, 4))
plt.plot(x_axis, log_kll_log, ".-", label="K-means||")
plt.plot(x_axis, log_akll_log, ".-", label="Accelerated K-Means||")
plt.xlabel("K")
plt.ylabel("log2(sec)")
plt.title(f"Runtime {dataset_name}")
plt.legend(loc="best")
plt.savefig(
    f"{root_dir}/report/{uid}_kmeans_parallel_{dataset_name}_Runtime_k_{start_ra
)
plt.show()

speedup_result = np.divide(kll_log, akll_log)
log_speedup_result = np.log(speedup_result)
plt.figure(facecolor="white", figsize=(6, 4))
plt.plot(x_axis, log_speedup_result, ".-", label='speedup')
plt.xlabel("K")
plt.ylabel("log2(Speedup)")
plt.title(f"Speed comparison {dataset_name}")
plt.legend(loc="best")
plt.savefig(
    f"{root_dir}/report/{uid}_kmeans_parallel_{dataset_name}_Speed_comparison_k_
)
plt.show()

df = pd.DataFrame(
    np.array(
        [
            kll_log,
            akll_log,
            speedup_result,
            log_kll_log,
            log_akll_log,
            log_speedup_result,
        ]
    ).T,
    columns=[
        "kll_log",
        "akll_log",
        "speedup_result",
        "log_kll_log",
        "log_akll_log",
        "log_speedup_result",
    ],
)
df.to_csv(
    f"{root_dir}/report/{uid}_kmeans_parallel_{dataset_name}_log_k_{start_range}
    index=False,
    encoding="utf-8-sig",
)
print("pause")
```

```
func:Dataset.__init__ took: 0.07171964645385742 sec
running kll for k: 32
func:Dataset.__init__ took: 1.3113021850585938e-05 sec
func:KPP.fit took: 0.0067098140716552734 sec
```

```
func:KLL.fit took: 74.30525660514832 sec
kll runs in: 74.3053252696991sec
running akll for k: 32
func:Dataset.__init__ took: 1.1205673217773438e-05 sec
func:KPP.fit took: 0.006355762481689453 sec
func:AKLL.fit took: 46.80021619796753 sec
akll runs in: 46.80028772354126sec
running kll for k: 64
func:Dataset.__init__ took: 1.1682510375976562e-05 sec
func:KPP.fit took: 0.01385951042175293 sec
func:KLL.fit took: 139.4877271652217 sec
kll runs in: 139.48809123039246sec
running akll for k: 64
func:Dataset.__init__ took: 1.5974044799804688e-05 sec
func:KPP.fit took: 0.013992786407470703 sec
func:AKLL.fit took: 89.14439058303833 sec
akll runs in: 89.14482498168945sec
running kll for k: 128
func:Dataset.__init__ took: 1.430511474609375e-05 sec
func:KPP.fit took: 0.04140782356262207 sec
func:KLL.fit took: 265.90474486351013 sec
kll runs in: 265.9048275947571sec
running akll for k: 128
func:Dataset.__init__ took: 1.4543533325195312e-05 sec
func:KPP.fit took: 0.044092416763305664 sec
func:AKLL.fit took: 192.26664972305298 sec
akll runs in: 192.2667465209961sec
running kll for k: 256
func:Dataset.__init__ took: 1.3828277587890625e-05 sec
func:KPP.fit took: 0.14312410354614258 sec
func:KLL.fit took: 536.2598567008972 sec
kll runs in: 536.260124206543sec
running akll for k: 256
func:Dataset.__init__ took: 1.4066696166992188e-05 sec
func:KPP.fit took: 0.1421186923980713 sec
func:AKLL.fit took: 320.96492075920105 sec
akll runs in: 320.96527433395386sec
running kll for k: 512
func:Dataset.__init__ took: 1.2159347534179688e-05 sec
func:KPP.fit took: 0.5066385269165039 sec
func:KLL.fit took: 1115.263266801834 sec
kll runs in: 1115.263375043869sec
running akll for k: 512
func:Dataset.__init__ took: 1.5735626220703125e-05 sec
func:KPP.fit took: 0.5123639106750488 sec
func:AKLL.fit took: 555.0675053596497 sec
akll runs in: 555.068021774292sec
running kll for k: 1024
func:Dataset.__init__ took: 1.52587890625e-05 sec
func:KPP.fit took: 3.7164103984832764 sec
func:KLL.fit took: 2057.9835138320923 sec
kll runs in: 2057.984004020691sec
running akll for k: 1024
func:Dataset.__init__ took: 1.3113021850585938e-05 sec
func:KPP.fit took: 2.118255853652954 sec
func:AKLL.fit took: 824.5932693481445 sec
akll runs in: 824.5938096046448sec
running kll for k: 2048
func:Dataset.__init__ took: 1.4781951904296875e-05 sec
func:KPP.fit took: 13.240400075912476 sec
```

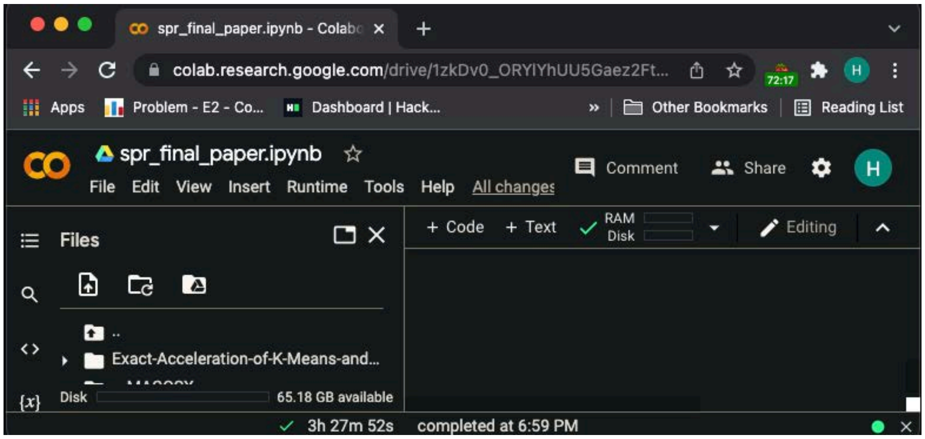
```
func:KLL.fit took: 5386.974714756012 sec
kll runs in: 5386.975218772888sec
running akll for k: 2048
func:Dataset.__init__ took: 1.5020370483398438e-05 sec
func:KPP.fit took: 9.787511825561523 sec
func:AKLL.fit took: 866.3631222248077 sec
akll runs in: 866.3632161617279sec
```





```
pause
```

```
Takes 3h 27m 52s
```



✓ 3h 27m 52s completed at 6:59 PM