

1. Налаштування середовища розробки:

- Налаштуйте середовище розробки для використання Vertex AI.

```
[2]: project = !gcloud config get-value project
PROJECT_ID = project[0]
PROJECT_ID
```

```
[2]: 'nk-cloud-course-iav-21'
```

```
[4]: # Ресурси
DEPLOY_COMPUTE = 'n1-standard-4'

REGION = 'us-central1'
DATASET = 'nk-cloud-course-iav-21'
DATABASE = 'nk_fraud_detection'
TABLE = 'transactions'
NOTEBOOK = '02a'

# Model Training
VAR_TARGET = 'isFraud'
VAR_OMIT = 'transaction_id nameOrig nameDest isFlaggedFraud splits' # додайте більше змінних через пробіл
```

пакекти:

```
[5]: from google.cloud import aiplatform
from datetime import datetime

from google.cloud import bigquery
from google.protobuf import json_format
from google.protobuf.struct_pb2 import Value
import json
import numpy as np
```

клієнти:

```
[6]: aiplatform.init(project=PROJECT_ID, location=REGION)
bigquery = bigquery.Client()
```

параметри:

```
[7]: TIMESTAMP = datetime.now().strftime("%Y%m%d%H%M%S")
DIR = f"temp/{NOTEBOOK}"
```

середовище:

```
[8]: !rm -rf {DIR}
!mkdir -p {DIR}
```

Створити набір даних (посилання на таблицю BigQuery)

```
[12]: dataset = aiplatform.TabularDataset.create(
    display_name = f'{NOTEBOOK}_{DATASET}_{TIMESTAMP}',
    bq_source = f'bq://{PROJECT_ID}.{DATABASE}.{TABLE}_prepped',
    labels = {'notebook': f'{NOTEBOOK}'}
)
```

Creating TabularDataset
Create TabularDataset backing LRO: projects/186934415911/locations/us-central1/datasets/6061138112464027648/operations/7054034530100838400
TabularDataset created. Resource name: projects/186934415911/locations/us-central1/datasets/6061138112464027648
To use this TabularDataset in another session:
ds = aiplatform.TabularDataset('projects/186934415911/locations/us-central1/datasets/6061138112464027648')

Навчання моделі з AutoML

```
[13]: column_specs = list(set(dataset.column_names) - set(VAR_OMIT.split()) - set([VAR_TARGET, 'splits']))
```

```
[14]: column_specs = dict.fromkeys(column_specs, 'auto')
```

Визначте job:

- Consider Weighting
- Model Type
- Optimization Objective

<https://cloud.google.com/python/docs/reference/aiplatform/latest/index.html#google.cloud.aiplatform.AutoMLTabularTrainingJob>

```
[15]: tabular_classification_job = aiplatform.AutoMLTabularTrainingJob(
    display_name = f'{NOTEBOOK}_{DATANAME}_{TIMESTAMP}',
    optimization_prediction_type = 'classification',
    optimization_objective = 'maximize-au-prc',
    column_specs = column_specs,
    labels = {'notebook': f'{NOTEBOOK}'}
)
```

```
[*]: model = tabular_classification_job.run(
    dataset = dataset,
    target_column = VAR_TARGET,
    predefined_split_column_name = 'splits',
    # training_fraction_split = 0.8,
    # validation_fraction_split = 0.1,
    # test_fraction_split = 0.1,
    budget_milli_node_hours = 1000,
    model_display_name = f'{NOTEBOOK}_{DATANAME}_{TIMESTAMP}',
    disable_early_stopping = False,
    model_labels = {'notebook': f'{NOTEBOOK}'}
)
```

View Training:

<https://console.cloud.google.com/ai/platform/locations/us-central1/training/768714792826306560?project=186934415911>

AutoMLTabularTrainingJob projects/186934415911/locations/us-central1/trainingPipelines/768714792826306560 current state:

PipelineState.PIPELINE_STATE_RUNNING

Filter Enter a property name											
Name	ID	Status	Job type	Model type	Duration	Last updated	Created	Ended	Labels		
02a_nk_fraud_detection_20240411110444	768714792826306560	Training	Training pipeline	Tabular classification	45 sec	Apr 11, 2024, 2:07:09 PM	Apr 11, 2024, 2:07:09 PM	—	notebook: 02a		

2. Побудова моделей з використанням Kubeflow:

- Ініціалізуйте конвеєр для навчання моделі з використанням Kubeflow.

Налаштування

ВХІДНІ ДАНІ:

```
[1]: project = !gcloud config get-value project
PROJECT_ID = project[0]
PROJECT_ID
```

```
[1]: 'nk-cloud-course-iaav-21'
```

```
[2]: # Перуcpu
DEPLOY_COMPUTE = 'n1-standard-2'

REGION = 'us-central1'
DATASET = 'nk-cloud-course-iaav-21'
DATABASE = 'nk_fraud_detection'
TABLE = 'transactions'
NOTEBOOK = '02a'
DATANAME = DATABASE

# Model Training
VAR_TARGET = 'isFraud'
VAR_OMIT = 'transaction_id nameOrig nameDest isFlaggedFraud splits' # додайте більше змінних через пробіл
```

ПАКЕТИ:

```
[4]: !pip install -U google-cloud-pipeline-components -U -q
```

```
[3]: from google.cloud import aiplatform
from datetime import datetime
import kfp
from kfp.v2 import compiler
#import kfp.v2.dsl as dsl
#import google_cloud_pipeline_components as gcc_aip
from google_cloud_pipeline_components.v1.dataset import TabularDatasetCreateOp
from google_cloud_pipeline_components.v1.automl.training_job import AutoMLTabularTrainingJobRunOp
from google_cloud_pipeline_components.v1.endpoint import EndpointCreateOp, ModelDeployOp

from google.cloud import bigquery
from google.protobuf import json_format
from google.protobuf.struct_pb2 import Value
import json
import numpy as np
```

/var/tmp/ipykernel_45258/450478126.py:4: DeprecationWarning: The module 'kfp.v2' is deprecated and will be removed in a future version. Please import directly from the 'kfp' namespace, instead of 'kfp.v2'.

КЛІЄНТИ:

```
[5]: aiplatform.init(project=PROJECT_ID, location=REGION)
bq = bigquery.Client()
```

параметри:

```
[6]: TIMESTAMP = datetime.now().strftime("%Y%m%d%H%M%S")
BUCKET = PROJECT_ID
URI = f"gs://{BUCKET}/{DATANAME}/models/{NOTEBOOK}"
DIR = f"temp/{NOTEBOOK}"
```

```
[7]: SERVICE_ACCOUNT = !gcloud config list --format='value(core.account)'\nSERVICE_ACCOUNT = SERVICE_ACCOUNT[0]\nSERVICE_ACCOUNT
```

```
[7]: '186934415911-compute@developer.gserviceaccount.com'
```

Перелік поточних службових акаунтів:

```
[8]: !gcloud projects get-iam-policy $PROJECT_ID --filter="bindings.members:$SERVICE_ACCOUNT" --format='table(bindings.role)' --flatten="bindings[].members"

ROLE
roles/editor
roles/storage.objectAdmin
```

Примітка: Якщо у отриманому переліку відсутній `roles/storage.objectAdmin`, тоді поверніться до блокнота налаштувань і додайте цей дозвіл до облікового запису, дотримуючись наданих інструкцій.

ОТОЧЕННЯ:

```
[9]: !rm -rf {DIR}
!mkdir -p {DIR}
```

Визначення Pipeline (KFP)

- Поточний шлях
 - Створення набору даних Vertex AI з посилання на таблицю BigQuery
 - Створення job для навчання табличних даних Vertex AI AutoML
 - Створіть Endpoint та розгорніть модель

Використання [AI Platform Pipeline Components](#)

- Особливо, [AutoMLTabularTrainingJobRunOp](#)

Визначення Job:

- Consider Weighting
- Model Type
- Optimization Objective

```
10]: @kfp.dsl.pipeline(  
    name = f'kfp-{NOTEBOOK}-{DATANAME}-{TIMESTAMP}',  
    pipeline_root = URI+'/' +str(TIMESTAMP)+' /kfp/'  
)  
def pipeline(  
    project: str,  
    dataname: str,  
    display_name: str,  
    deploy_machine: str,  
    bq_source: str,  
    var_target: str,  
    var_omit: str,  
    features: dict,  
    labels: dict  
):  
  
    # dataset  
    dataset = TabularDatasetCreateOp(  
        project = project,  
        display_name = display_name,  
        bq_source = bq_source,  
        labels = labels  
    )  
  
    # training  
    model = AutoMLTabularTrainingJobRunOp(  
        project = project,  
        display_name = display_name,  
        optimization_prediction_type = "classification",  
        optimization_objective = "maximize-au-prc",  
        budget_milli_node_hours = 1000,  
        disable_early_stopping=False,  
        column_specs = features,  
        dataset = dataset.outputs['dataset'],  
        target_column = var_target,  
        predefined_split_column_name = 'splits',
```

```

        labels = labels
    )

    # Endpoint: створення
    endpoint = EndpointCreateOp(
        project = project,
        display_name = display_name,
        labels = labels
    )

    # Endpoint: розгортання моделі
    deployment = ModelDeployOp(
        model = model.outputs["model"],
        endpoint = endpoint.outputs["endpoint"],
        dedicated_resources_min_replica_count = 1,
        dedicated_resources_max_replica_count = 1,
        traffic_split = {"0": 100},
        dedicated_resources_machine_type= deploy_machine
    )

```

- Скопіюйте ініціалізований конвеєр.

Компіляція Pipeline

```

[11]: compiler.Compiler().compile(
        pipeline_func = pipeline,
        package_path = f"{DIR}/{NOTEBOOK}.json"
    )

```

Перемістіть скопійовані файли Pipeline до GCS Bucket

```

[12]: !gsutil cp {DIR}/{NOTEBOOK}.json {URI}/{TIMESTAMP}/kfp/

```

```

Copying file://temp/02a/02a.json [Content-Type=application/json]...
- [1 files][ 46.6 KiB/ 46.6 KiB]
Operation completed over 1 objects/46.6 KiB.

```

- Створіть Vertex AI Pipeline Job.

Створення Vertex AI Pipeline Job

Отримати словник ознак для вхідних даних конвеєра:

```

[16]: # отримання назв змінних
query = f"SELECT * FROM {DATANAME}.INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME = '{TABLE}_prepped'"
schema = bq.query(query).to_dataframe()
OMIT = VAR_OMIT.split() + [VAR_TARGET, 'splits']
features = schema[~schema.column_name.isin(OMIT)].column_name.tolist()
features = dict.fromkeys(features, 'auto')

```

Запуск pipeline:

```

[18]: pipeline = aiplatform.PipelineJob(
        display_name = f'{NOTEBOOK}_{DATANAME}_{TIMESTAMP}',
        template_path = f'{URI}/{TIMESTAMP}/kfp/{NOTEBOOK}.json",
        parameter_values = {
            "project" : PROJECT_ID,
            "dataname" : DATANAME,
            "display_name" : f'{NOTEBOOK}_{DATANAME}_{TIMESTAMP}',
            "deploy_machine" : DEPLOY_COMPUTE,
            "bq_source" : f'bq://{PROJECT_ID}.{DATANAME}.{TABLE}_prepped',
            "var_target" : VAR_TARGET,
            "var_omit" : VAR_OMIT,
            "features" : features,
            "labels" : {'notebook': NOTEBOOK}
        },
        labels = {'notebook': NOTEBOOK},
        enable_caching=False
    )

[*]: response = pipeline.run(
        service_account = SERVICE_ACCOUNT
    )

```

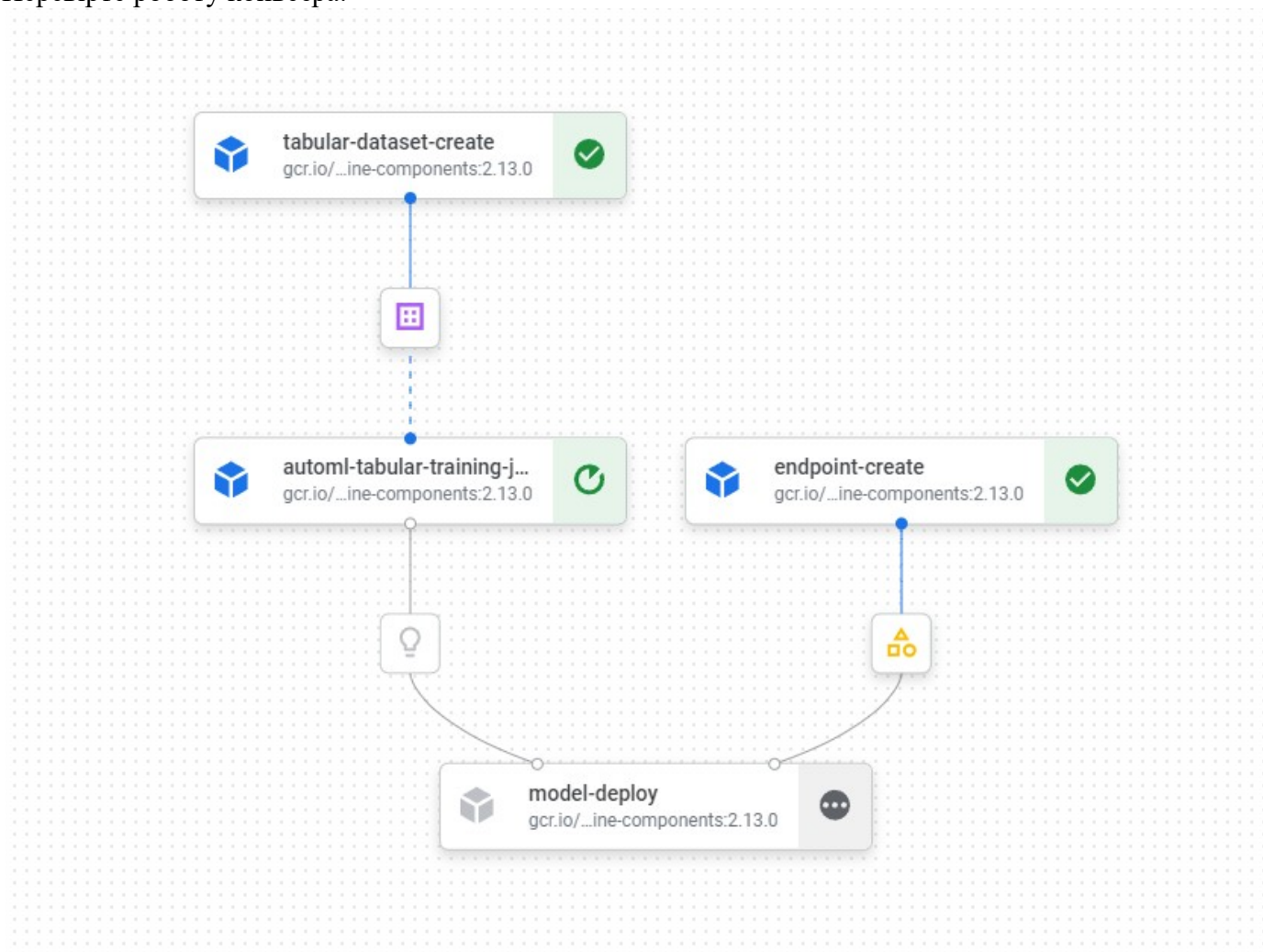
```

Creating PipelineJob
PipelineJob created. Resource name: projects/186934415911/locations/us-central1/pipelineJobs/kfp-02a-nk-fraud-detection-20240411110925-20240411111623
To use this PipelineJob in another session:
pipeline_job = aiplatform.PipelineJob.get('projects/186934415911/locations/us-central1/pipelineJobs/kfp-02a-nk-fraud-detection-20240411110925-20240411111623')
View Pipeline Job:
https://console.cloud.google.com/vertex-ai/locations/us-central1/pipelines/runs/kfp-02a-nk-fraud-detection-20240411110925-20240411111623?project=186934415911
PipelineJob projects/186934415911/locations/us-central1/pipelineJobs/kfp-02a-nk-fraud-detection-20240411110925-20240411111623 current state:
PipelineState.PIPELINE_STATE_RUNNING

```

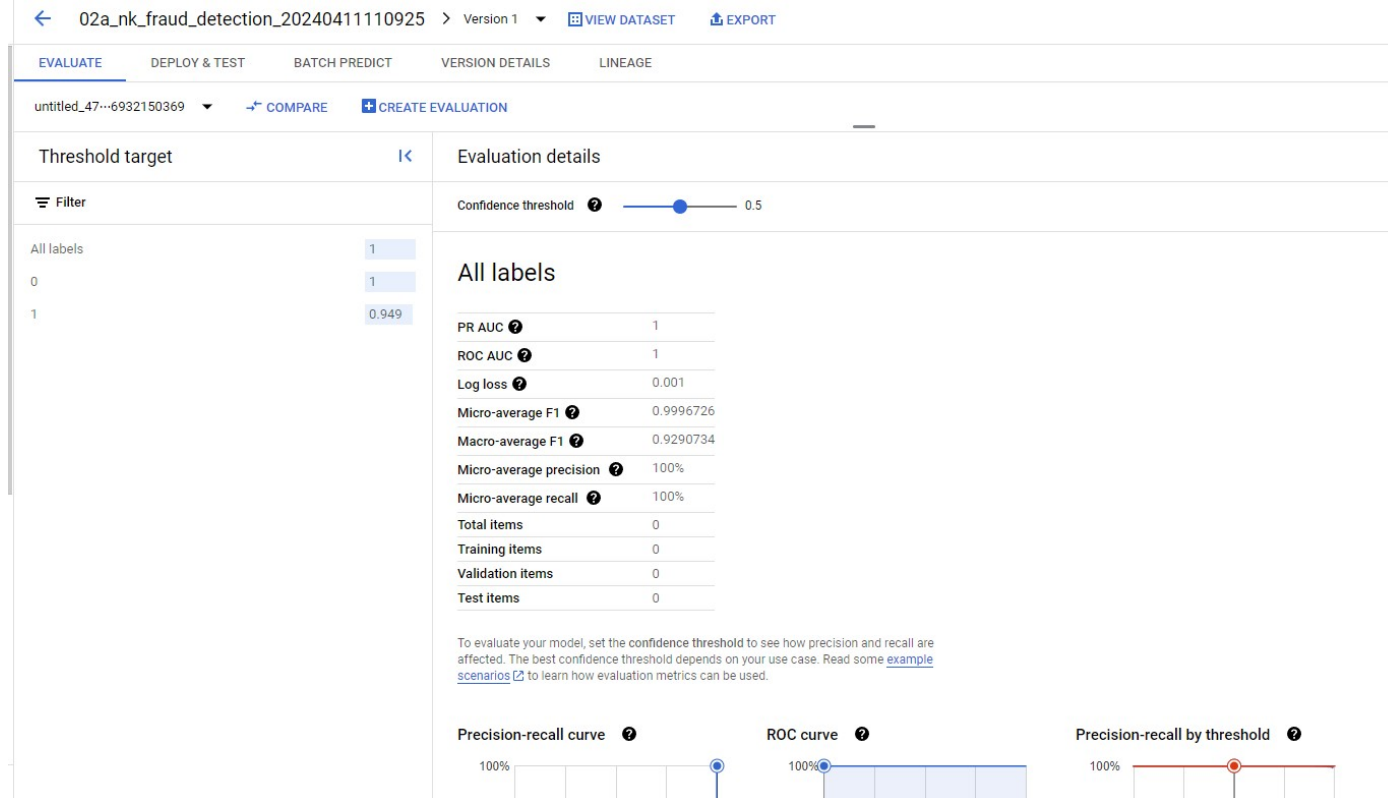

Filter	Filter runs									
Run	Status	Pipeline / Component	Duration	Created ↓	Ended	Experiment	Experiment run	Labels ?		
<input type="checkbox"/> kfp-02a-nk-fraud-detection-20240411110925-20240411111056	Running	kfp-02a-nk-fraud-detection-20240411110925	2 min 45 sec	Apr 11, 2024, 2:10:58 PM				notebook: 02a		

- Перевірте роботу конвеєра.



3. Оцінка якості та отримання прогнозних значень:

- Оцініть якість моделі за допомогою метрик якості.



- Порівняйте якість моделі з попередніми результатами.

```
[25]: evaluation.keys()

[25]: dict_keys(['name', 'metricsSchemaUri', 'metrics', 'createTime', 'sliceDimensions', 'modelExplanation'])

[26]: evaluation['metrics'].keys()

[26]: dict_keys(['auPrc', 'auRoc', 'logLoss', 'confusionMatrix', 'confidenceMetrics'])

[27]: evaluation['metrics']['auPrc']

[27]: 0.9999913

[28]: evaluation['metrics']['confidenceMetrics'][3]

[28]: {'falseNegativeCount': '4',
      'recallAt1': 0.9996726,
      'falsePositiveRateAt1': 0.000327409,
      'truePositiveCount': '635287',
      'recall': 0.9999937,
      'trueNegativeCount': '631493',
      'confidenceThreshold': 0.01,
      'falsePositiveRate': 0.0059783626,
      'precisionAt1': 0.9996726,
      'confusionMatrix': {'rows': [[634453.0, 10.0, 0.0],
                                   [198.0, 630.0, 0.0],
                                   [0.0, 0.0, 0.0]]},
      'annotationSpecs': [{'displayName': '0', 'id': '0'},
                          {'displayName': '1', 'id': '1'},
                          {'displayName': 'DROPPED', 'id': 'DROPPED'}]},
      'precision': 0.9940571,
      'f1ScoreAt1': 0.9996726,
      'f1ScoreMacro': 0.6689364,
      'f1Score': 0.99701655,
      'falsePositiveCount': '3798',
      'f1ScoreMicro': 0.99701655}
```

Перегляньте кілька показників. Також порівняйте їх з результатами з консолі.

```
[29]: print(f"Переглянути цю модель в консолі: \nhttps://console.cloud.google.com/vertex-ai/locations/{REGION}/models/{model.name}/versions/{model.version_id}/evaluations/{evaluation['name']}.split('/')[-1]
<
Переглянути цю модель в консолі:
https://console.cloud.google.com/vertex-ai/locations/us-central1/models/8478977376082984960/versions/1/evaluations/476154246932150369?project=nk-cloud-course-ia-v-21

[30]: evaluation['metrics']['auPrc']

[30]: 0.9999913

[31]: for i in range(len(evaluation['metrics']['confusionMatrix']['annotationSpecs'])):
    print('True Label = ', evaluation['metrics']['confusionMatrix']['annotationSpecs'][i]['displayName'], ' has Predicted labels = ', evaluation['metrics']['confusionMatrix']['rows'][i])

True Label = 0 has Predicted labels = [634453.0, 10.0, 0.0]
True Label = 1 has Predicted labels = [198.0, 630.0, 0.0]
True Label = DROPPED has Predicted labels = [0.0, 0.0, 0.0]

Для моделей з мітками ви також можете отримати метрики оцінювання для кожного зрізу моделі, використовуючи версію .gcpic api:

[32]: model_client = aiplatform.gapic.ModelServiceClient(
    client_options = {
        'api_endpoint': f'{REGION}-aiplatform.googleapis.com'
    }
)

[33]: slices = model_client.list_model_evaluation_slices(parent = evaluation['name'])

[34]: for slice in slices:
    print('Label = ', slice.slice_value, ' has auPrc = ', slice.metrics['auPrc'])

Label = 0 has auPrc = 0.9999935
Label = 1 has auPrc = 0.9492538
```

- Отримайте прогнознi значення для тестового набору даних.

Прогнозування

Вхідні дані для прогнозування

```
[41]: query = f"""
        SELECT * EXCEPT({VAR_TARGET}, {'.'.join(VAR_OMIT.split(' '))})
        FROM {DATANAME}.{TABLE}_prepped
        WHERE splits='TEST'
        LIMIT 10
    """

query
pred = bq.query(query = query).to_dataframe()
```

```
[42]: pred.head(4)
```

```
[42]:
```

	step	type	amount	oldbalanceOrig	newbalanceOrig	oldbalanceDest	newbalanceDest
0	256	CASH_IN	159830.09	6025.00	165855.09	272867.99	113037.90
1	256	CASH_IN	292821.63	0.00	292821.63	7140417.91	6847596.28
2	256	CASH_IN	194707.03	8217217.95	8411924.97	307086.75	112379.72
3	256	CASH_IN	307824.44	755531.85	1063356.29	352047.67	44223.23

```
[44]: # pred['Time'] = pred['Time'].astype(str)
newobs = pred.to_dict(orient='records')
newobs[0]
```

```
[44]: {'step': 256,
      'type': 'CASH_IN',
      'amount': 159830.09,
      'oldbalanceOrig': 6025.0,
      'newbalanceOrig': 165855.09,
      'oldbalanceDest': 272867.99,
      'newbalanceDest': 113037.9}
```

Наступні комірки отримують модель з кінцевої точки та її схеми:

```
[45]: instances = [json_format.ParseDict(newob, Value()) for newob in newobs]
```


- Збережіть прогнозні значення у файл.

Отримання прогнозів: Python Client

```
[46]: aiplatform.Endpoint.list(filter=f'labels.notebook={NOTEBOOK}')
```

```
[46]: [<google.cloud.aiplatform.models.Endpoint object at 0x7f86010b2d00>
resource name: projects/186934415911/locations/us-central1/endpoints/8979832410325647360,
<google.cloud.aiplatform.models.Endpoint object at 0x7f86010b2fd0>
resource name: projects/186934415911/locations/us-central1/endpoints/1097407162520436736]
```

```
[47]: endpoint = aiplatform.Endpoint.list(filter=f'labels.notebook={NOTEBOOK}')[0]
endpoint.display_name
```

```
[47]: '02a_nk_fraud_detection_20240411110925'
```

```
[51]: prediction = endpoint.predict.instances = instances) # or instances = newobs
prediction.predictions[0]
```

```
[51]: {'classes': ['0', '1'], 'scores': [0.9995975494384766, 0.0004024186637252569]}
```

```
[52]: prediction.predictions[0]['classes'][np.argmax(prediction.predictions[0]['scores'])]
```

```
[52]: '0'
```

← 02a_nk_fraud_detection_20240411110925

ID	3273419873086078976
Model	02a_nk_fraud_detection_20240411110925 (Version 1)
Objective	Tabular
Import location	bq://nk-cloud-course-iav-21.nk_fraud_detection.transactions_prepped
Created	Apr 11, 2024 at 05:02PM
Updated	Apr 11, 2024 at 05:02PM
Elapsed time	13 sec
Status	Pending
Logs	View Logs
Export location	
Encryption type	Google-managed

- Завантажте файл з прогнозними значеннями у хмарне сховище.

<input type="checkbox"/>	forecast.json	411 B	application/json	Apr 11, 2024, 5:10:36 PM	Standard	Apr 11, 2024, 5:10:36 PM	Not public	
--------------------------	-------------------------------	-------	------------------	--------------------------	----------	--------------------------	------------	--

- Перевірте збережені прогнознi значення.

Автоматичне форматування ☐

```
{
  "predictions": [
    {
      "classes": [
        "0",
        "1"
      ],
      "scores": [
        0.99959754943847656,
        0.00040241866372525692
      ]
    }
  ],
  "deployedModelId": "2390740700040790016",
  "model": "projects/186934415911/locations/us-central1/models/8478977376082984960",
  "modelDisplayName": "02a_nk_fraud_detection_20240411110925",
  "modelVersionId": "1"
}
```

4. Звіт та висновки:

- Підготуйте звіт, що включає в себе опис процесу роботи, отримані результати, аналіз та висновки.