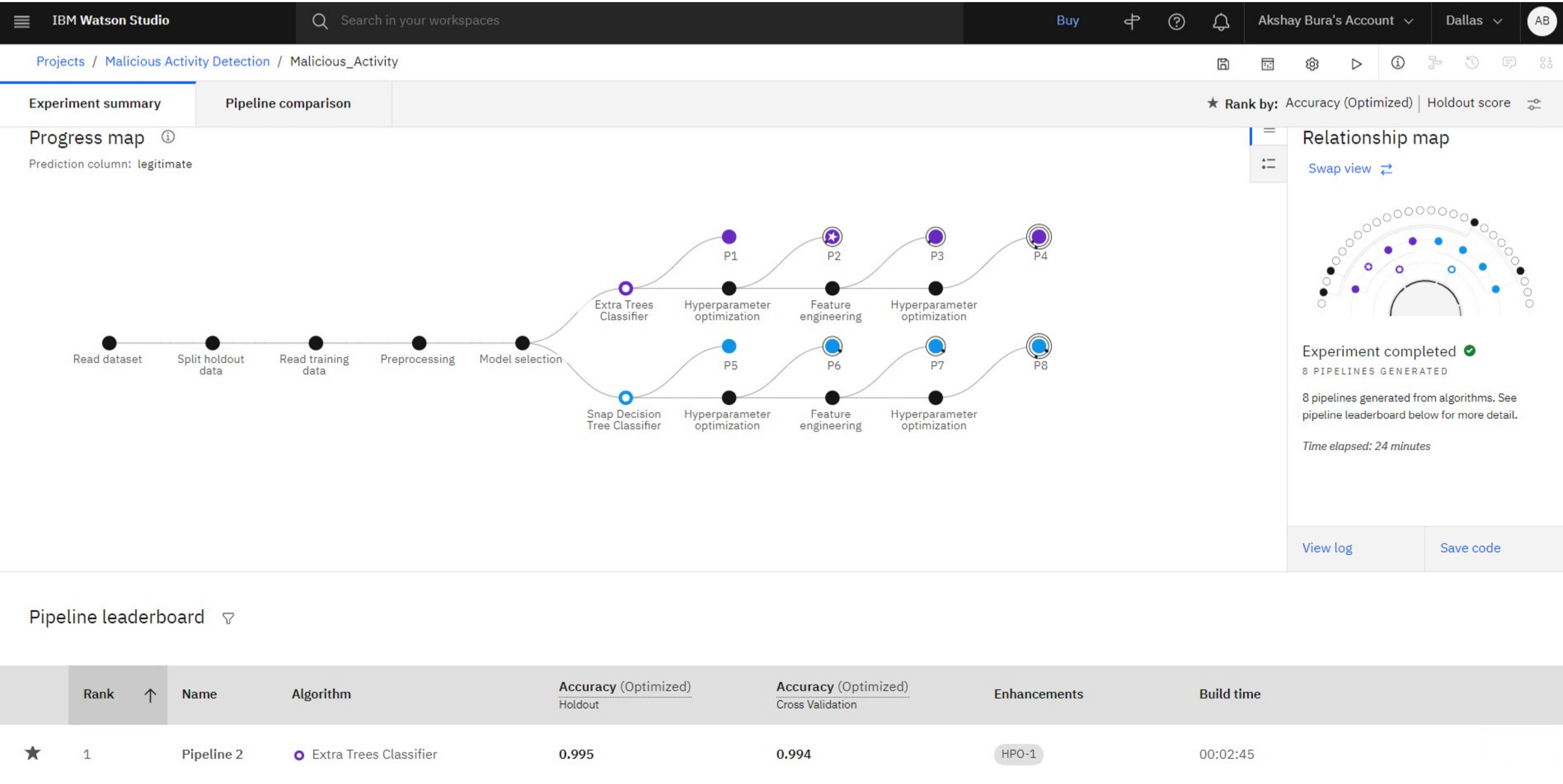


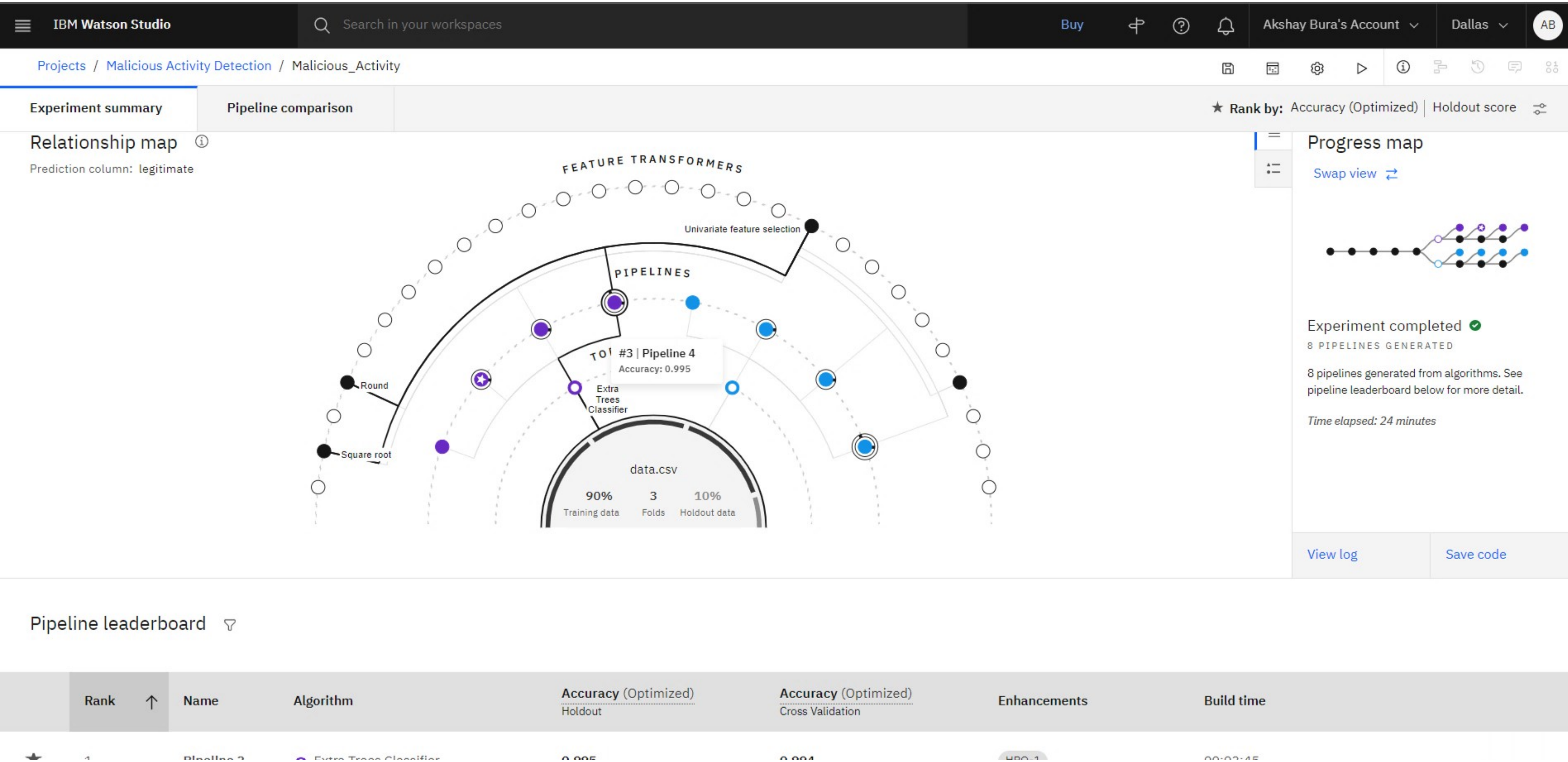
Experiment completed ✓

8 PIPELINES GENERATED

8 pipelines generated from algorithms. See pipeline leaderboard below for more detail.

Time elapsed: 24 minutes

	Rank ↑	Name	Algorithm	**Accuracy** (Optimized) Holdout	**Accuracy** (Optimized) Cross Validation	Enhancements	Build time
★	1	Pipeline 2	○ Extra Trees Classifier	**0.995**	**0.994**	HPO-1	00:02:45



Prediction column: legitimate

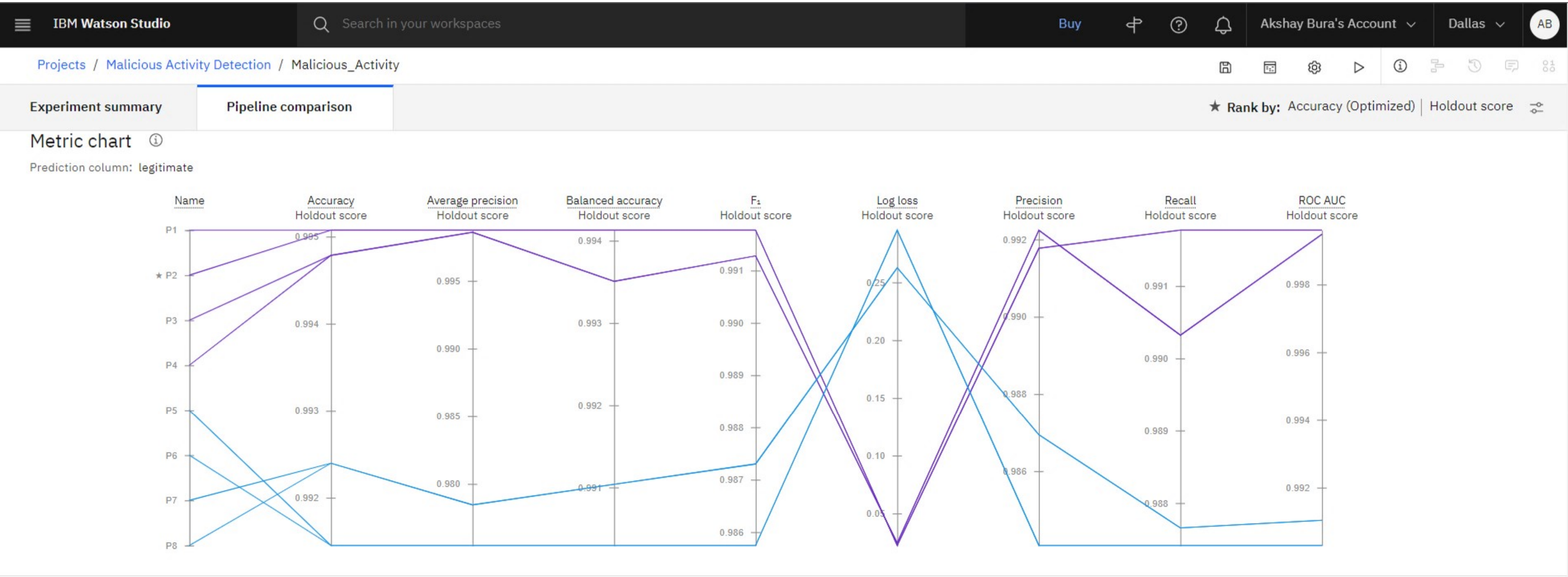
Experiment completed ✓

8 PIPELINES GENERATED

8 pipelines generated from algorithms. See pipeline leaderboard below for more detail.

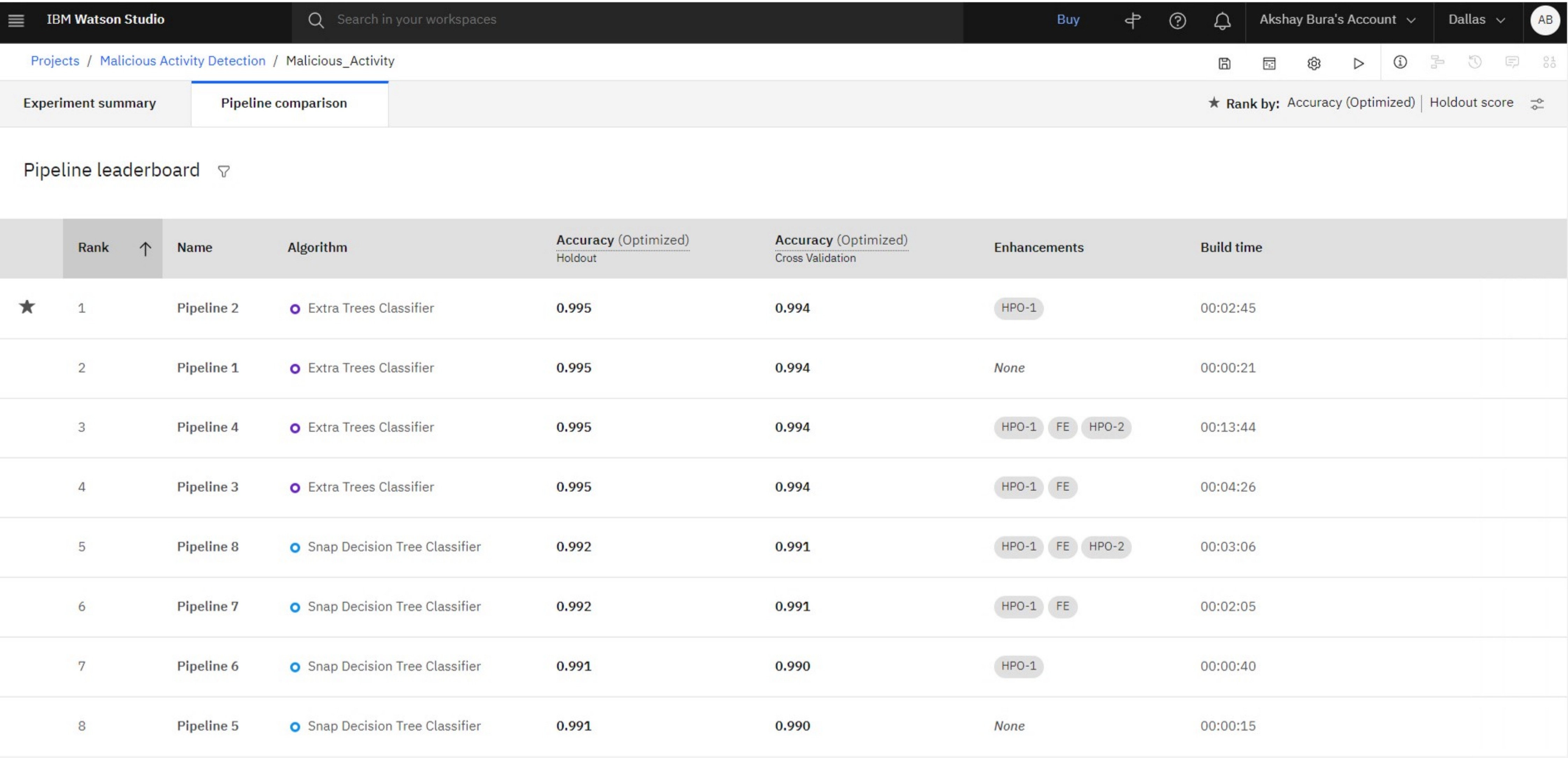
Time elapsed: 24 minutes

	Rank ↑	Name	Algorithm	Accuracy (Optimized) Holdout	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★	1	Pipeline 3	Extra Trees Classifier	0.995	0.994	HPO-1	00:02:45



Metric chart

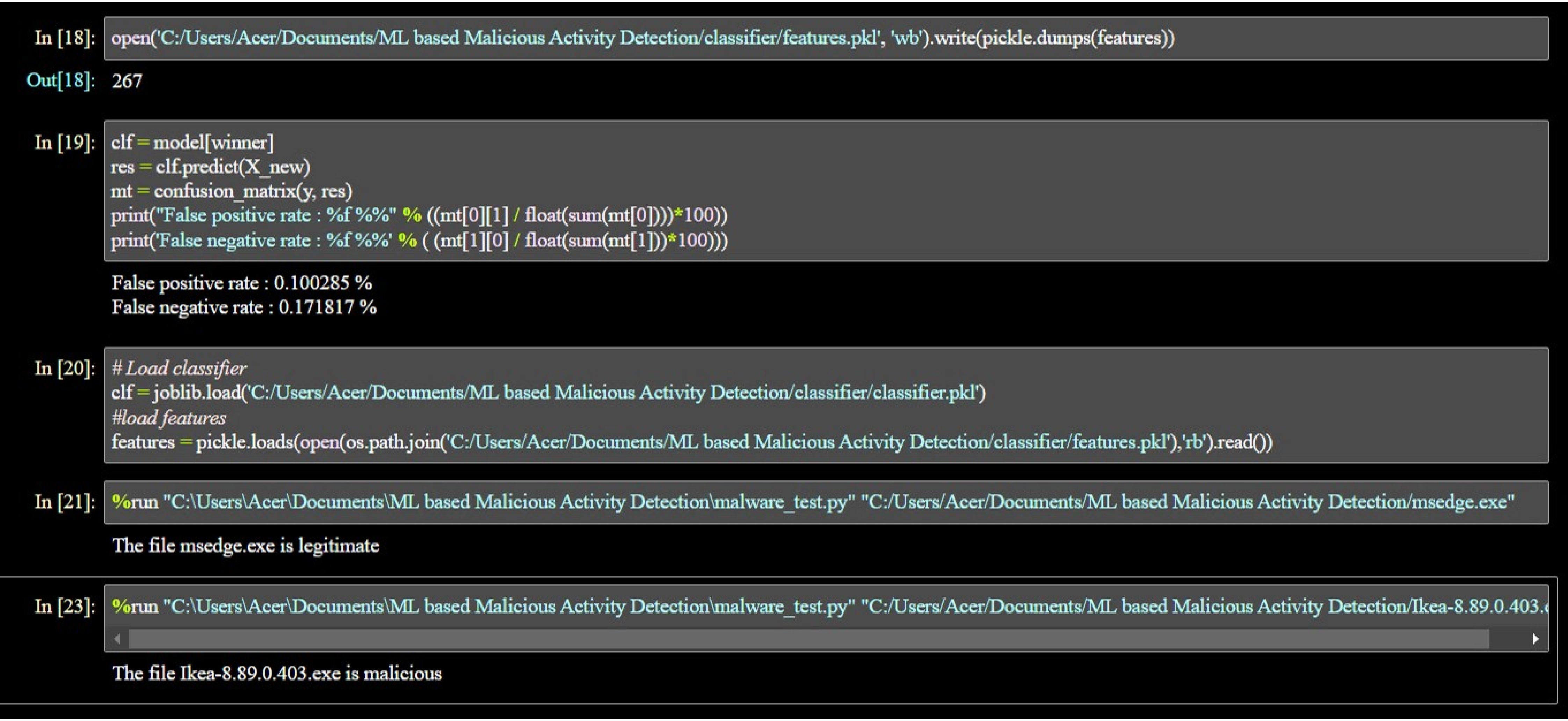
Prediction column: legitimate



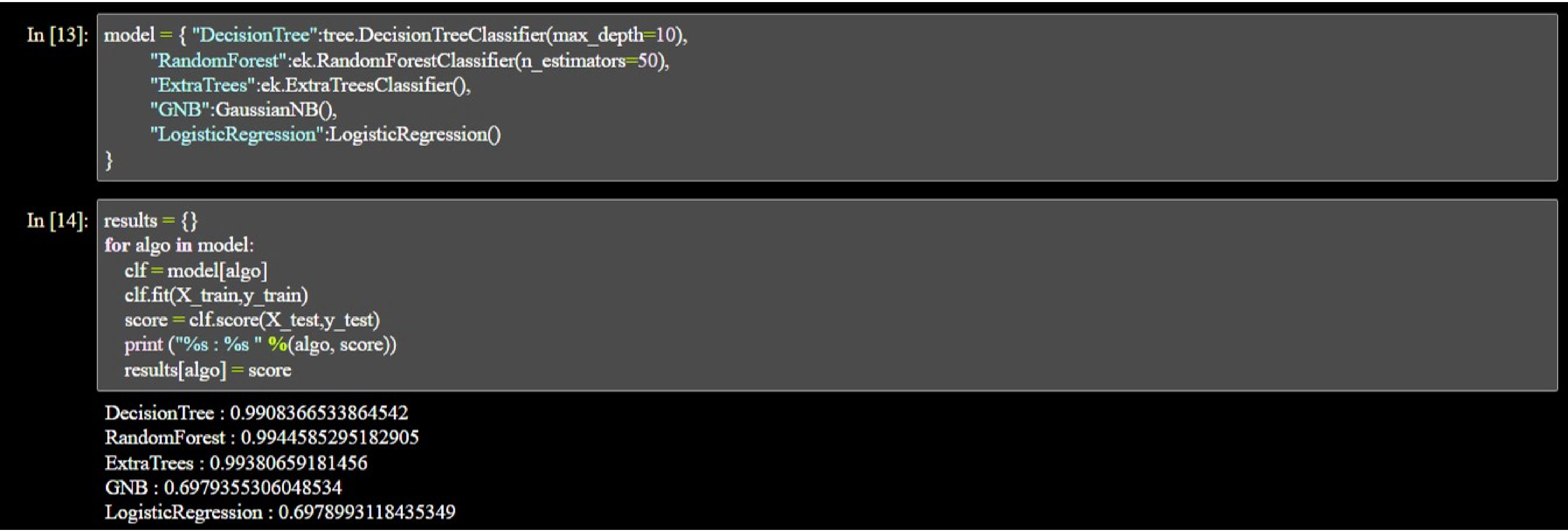
★ Rank by: Accuracy (Optimized) | Holdout score

Pipeline leaderboard

	Rank	Name	Algorithm	Accuracy (Optimized) Holdout	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★	1	Pipeline 2	○ Extra Trees Classifier	0.995	0.994	HPO-1	00:02:45
	2	Pipeline 1	○ Extra Trees Classifier	0.995	0.994	*None*	00:00:21
	3	Pipeline 4	○ Extra Trees Classifier	0.995	0.994	HPO-1 FE HPO-2	00:13:44
	4	Pipeline 3	○ Extra Trees Classifier	0.995	0.994	HPO-1 FE	00:04:26
	5	Pipeline 8	○ Snap Decision Tree Classifier	0.992	0.991	HPO-1 FE HPO-2	00:03:06
	6	Pipeline 7	○ Snap Decision Tree Classifier	0.992	0.991	HPO-1 FE	00:02:05
	7	Pipeline 6	○ Snap Decision Tree Classifier	0.991	0.990	HPO-1	00:00:40
	8	Pipeline 5	○ Snap Decision Tree Classifier	0.991	0.990	*None*	00:00:15



```
In [18]: open('C:/Users/Acer/Documents/ML based Malicious Activity Detection/classifier/features.pkl', 'wb').write(pickle.dumps(features))

Out[18]: 267

In [19]: clf = model[winner]
         res = clf.predict(X_new)
         mt = confusion_matrix(y, res)
         print("False positive rate : %f %%" % ((mt[0][1] / float(sum(mt[0])))*100))
         print('False negative rate : %f %%' % ( (mt[1][0] / float(sum(mt[1]))*100)))

         False positive rate : 0.100285 %
         False negative rate : 0.171817 %

In [20]: # Load classifier
         clf = joblib.load('C:/Users/Acer/Documents/ML based Malicious Activity Detection/classifier/classifier.pkl')
         #load features
         features = pickle.loads(open(os.path.join('C:/Users/Acer/Documents/ML based Malicious Activity Detection/classifier/features.pkl'),'rb').read())

In [21]: %run "C:\Users\Acer\Documents\ML based Malicious Activity Detection\malware_test.py" "C:/Users/Acer/Documents/ML based Malicious Activity Detection/msedge.exe"

         The file msedge.exe is legitimate

In [23]: %run "C:\Users\Acer\Documents\ML based Malicious Activity Detection\malware_test.py" "C:/Users/Acer/Documents/ML based Malicious Activity Detection/Ikea-8.89.0.403.

         The file Ikea-8.89.0.403.exe is malicious
```



```
In [13]: model = { "DecisionTree":tree.DecisionTreeClassifier(max_depth=10),
                 "RandomForest":ek.RandomForestClassifier(n_estimators=50),
                 "ExtraTrees":ek.ExtraTreesClassifier(),
                 "GNB":GaussianNB(),
                 "LogisticRegression":LogisticRegression()
         }
```

```
In [14]: results = {}
         for algo in model:
             clf = model[algo]
             clf.fit(X_train,y_train)
             score = clf.score(X_test,y_test)
             print ("%s : %s " %(algo, score))
             results[algo] = score
```

```
DecisionTree : 0.9908366533864542
RandomForest : 0.9944585295182905
ExtraTrees : 0.99380659181456
GNB : 0.6979355306048534
LogisticRegression : 0.6978993118435349
```