Eli Taylor

AI510

PE04

This assignment used Ludwig for AutoML. I found the library not as easy to work with as I had hoped. I had the same issue some other classmates also reported, related to importing torch. The assignment works in Google Colab for me. A benefit of the approach with Ludwig is that all the model building is abstracted away, so you can simply train and run a model without worrying about things like hyperparameter tuning. This makes the model easier to implement but also less transparent and interpretable.

```
from ludwig.api import LudwigModel

# import and train model
model = LudwigModel(config='model_definition.yaml')
model.train(dataset=df)
```

```
/usr/local/lib/python3.10/dist-packages/torch/nn/modules/conv.py:306: UserWarning: Using padding='same' with even kernel lengths ar
  return F.conv1d(input, weight, bias, self.stride,
```

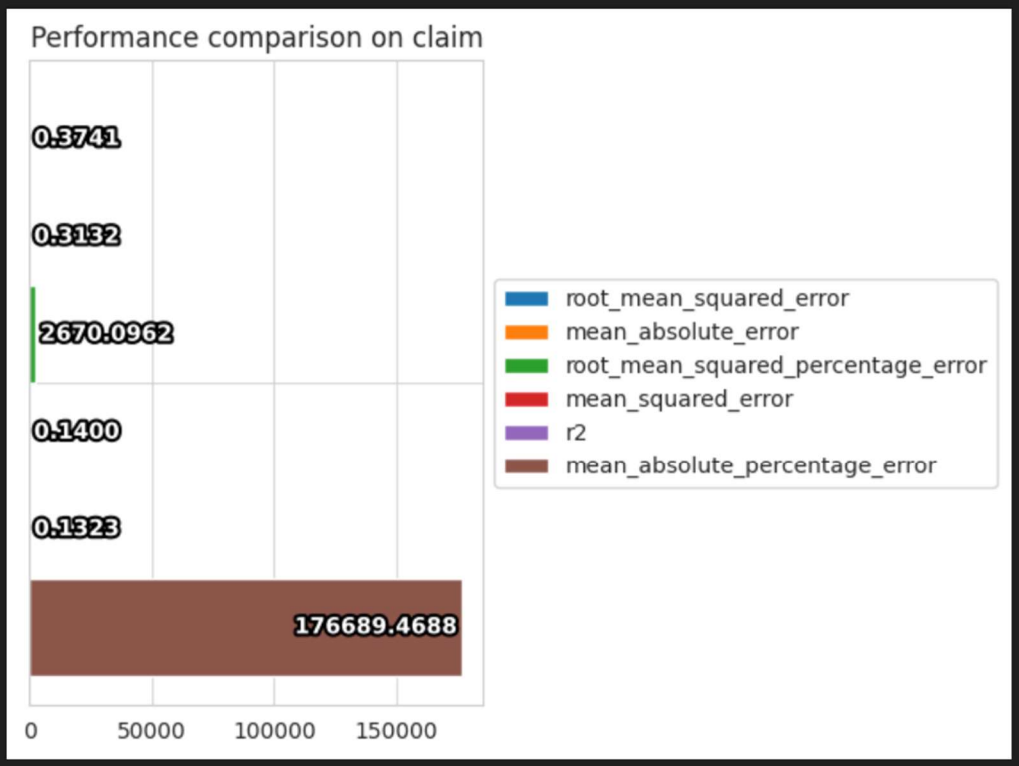
```
TrainingResults(train_stats=TrainingStats(training={'claim': {'loss': [16.162973403930664, 0.4036794602870941, 0.38176482915878296,
```

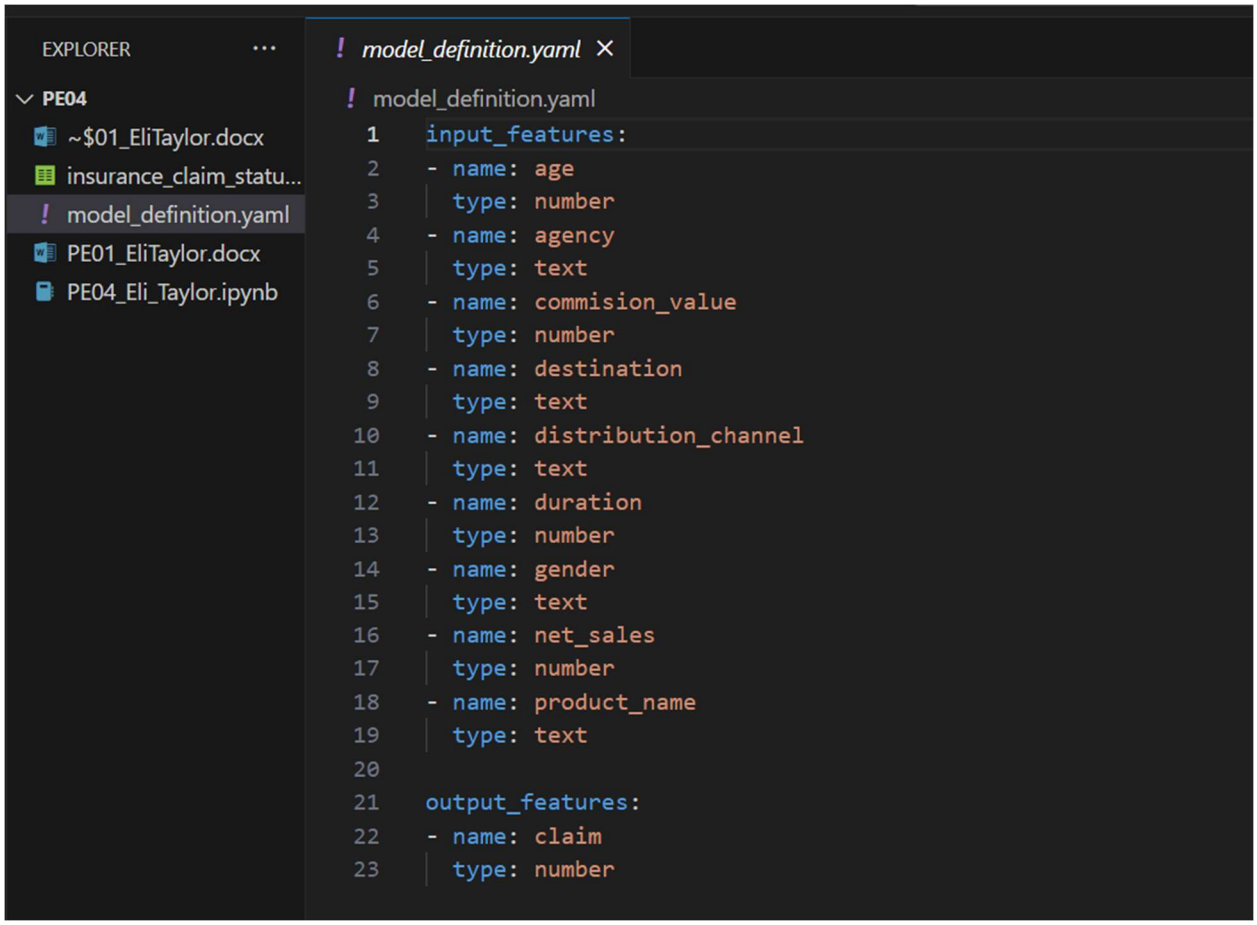
```
# get predictions
predictions, _ = model.predict(dataset='insurance_claim_status.csv')
predictions.head()
```

	claim_predictions
0	0.472934
1	0.032298
2	0.101622
3	0.127762
4	0.279444

```python
from ludwig.visualize import compare_performance

# evaluate model
eval_stats, _, _ = model.evaluate(dataset='insurance_claim_status.csv')
compare_performance([eval_stats])
```




EXPLORER
PE04
~$01_EliTaylor.docx
insurance_claim_statu...
model_definition.yaml
PE01_EliTaylor.docx
PE04_Eli_Taylor.ipynb
model_definition.yaml
model_definition.yaml
input_features:
- name: age
type: number
- name: agency
type: text
- name: commision_value
type: number
- name: destination
type: text
- name: distribution_channel
type: text
- name: duration
type: number
- name: gender
type: text
- name: net_sales
type: number
- name: product_name
type: text
output_features:
- name: claim
type: number