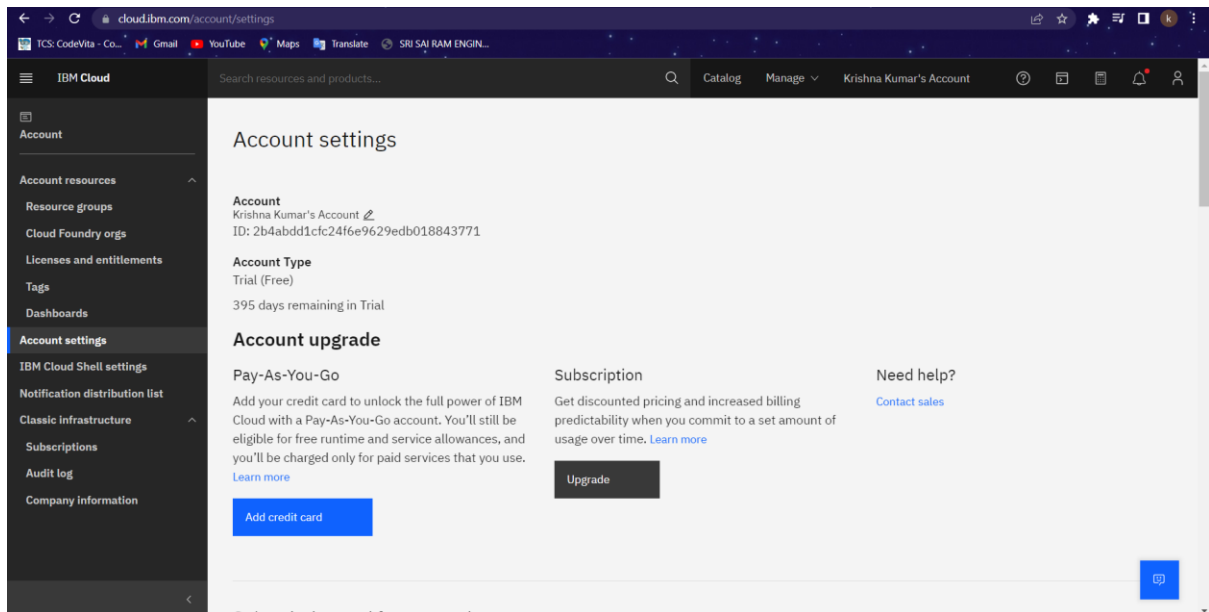


Training the model on IBM Watson Cloud:

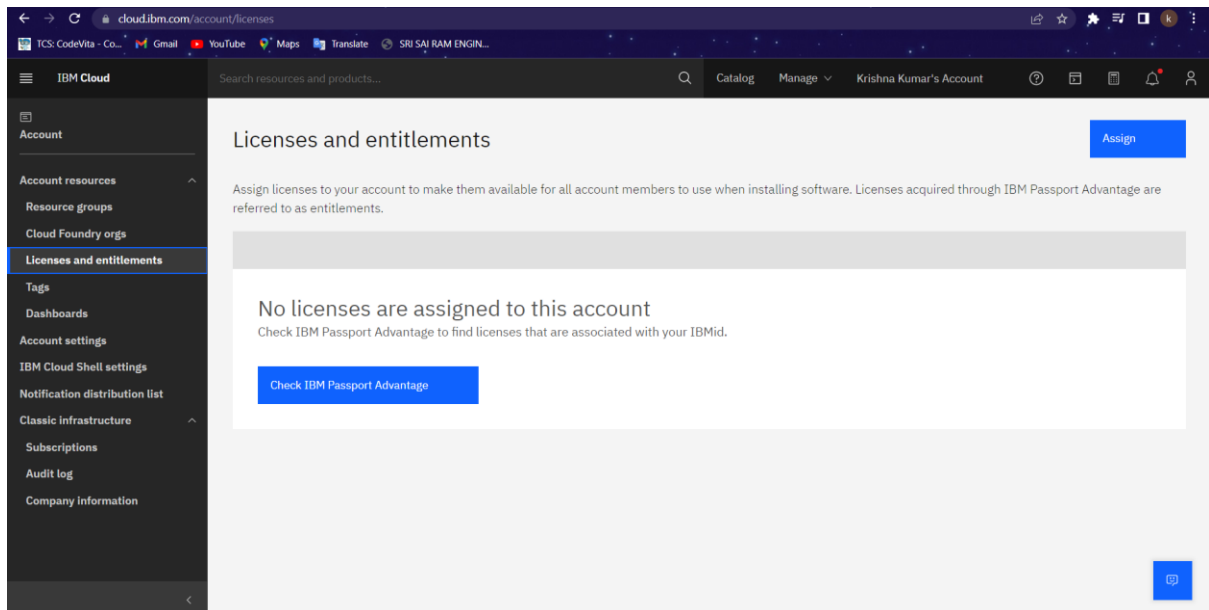
Creating account



The screenshot shows the IBM Cloud account settings page. The left sidebar contains a navigation menu with options: Account, Account resources, Resource groups, Cloud Foundry orgs, Licenses and entitlements, Tags, Dashboards, Account settings (highlighted), IBM Cloud Shell settings, Notification distribution list, Classic infrastructure, Subscriptions, Audit log, and Company information. The main content area is titled 'Account settings' and displays the following information:

- Account:** Krishna Kumar's Account, ID: 2b4abdd1cfc24f6e9629edb018843771
- Account Type:** Trial (Free), 395 days remaining in Trial
- Account upgrade:** Pay-As-You-Go (Add credit card button), Subscription (Upgrade button), and Need help? (Contact sales link)

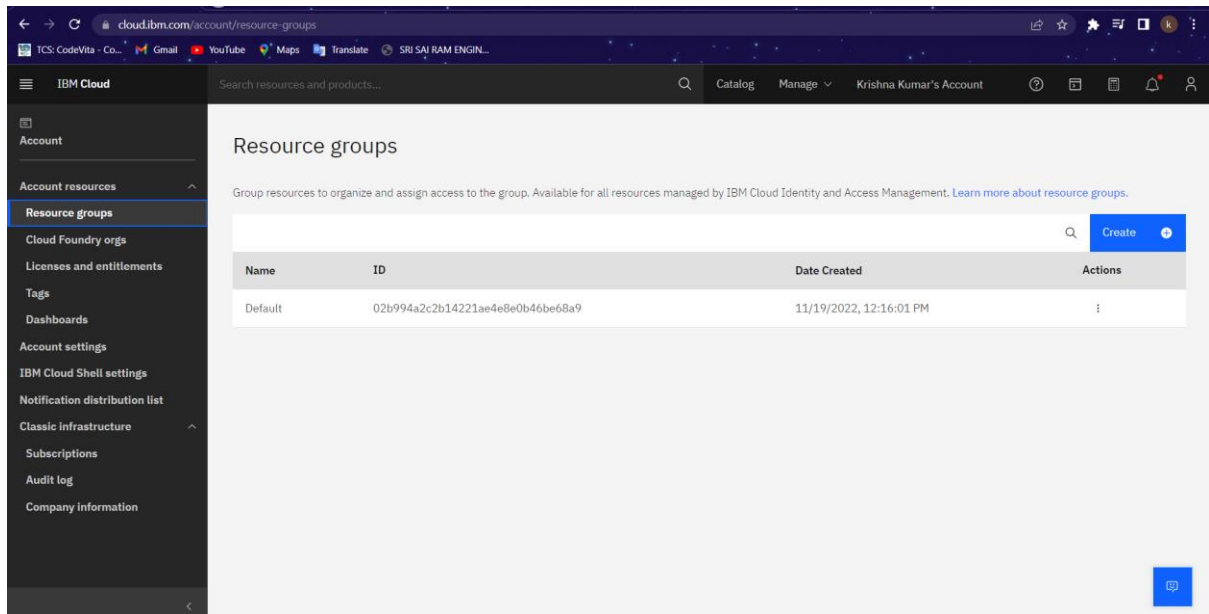
At the bottom, there is a section for 'Subscription and feature codes' with a blue button.



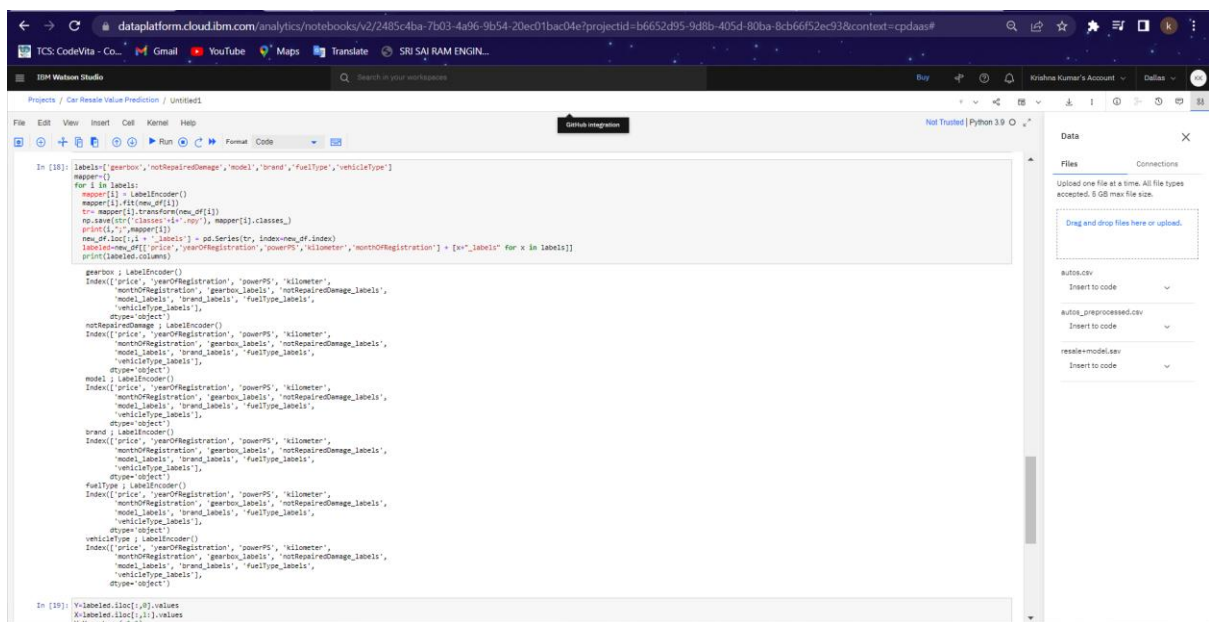
The screenshot shows the IBM Cloud 'Licenses and entitlements' page. The left sidebar is the same as the previous screenshot, with 'Licenses and entitlements' highlighted. The main content area is titled 'Licenses and entitlements' and includes an 'Assign' button. Below the title, it states: 'Assign licenses to your account to make them available for all account members to use when installing software. Licenses acquired through IBM Passport Advantage are referred to as entitlements.'

A message box indicates: 'No licenses are assigned to this account. Check IBM Passport Advantage to find licenses that are associated with your IBMid.' Below this message is a blue button labeled 'Check IBM Passport Advantage'.

Attaching resource



Training data



```
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import r2_score
regressor = RandomForestRegressor(n_estimators=1000, max_depth=10, random_state=34)

from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import r2_score
regressor = RandomForestRegressor(n_estimators=1000, max_depth=10, random_state=34)

regressor.fit(X_train, np.ravel(Y_train, order='C'))
Out[26]: RandomForestRegressor(max_depth=10, n_estimators=1000, random_state=34)

y_pred = regressor.predict(X_test)
print(r2_score(Y_test, y_pred))
0.834527626497731

filename = 'resale+model.sav'
pickle.dump(regressor, open(filename, 'wb'))

In [ ]:
```

Result

