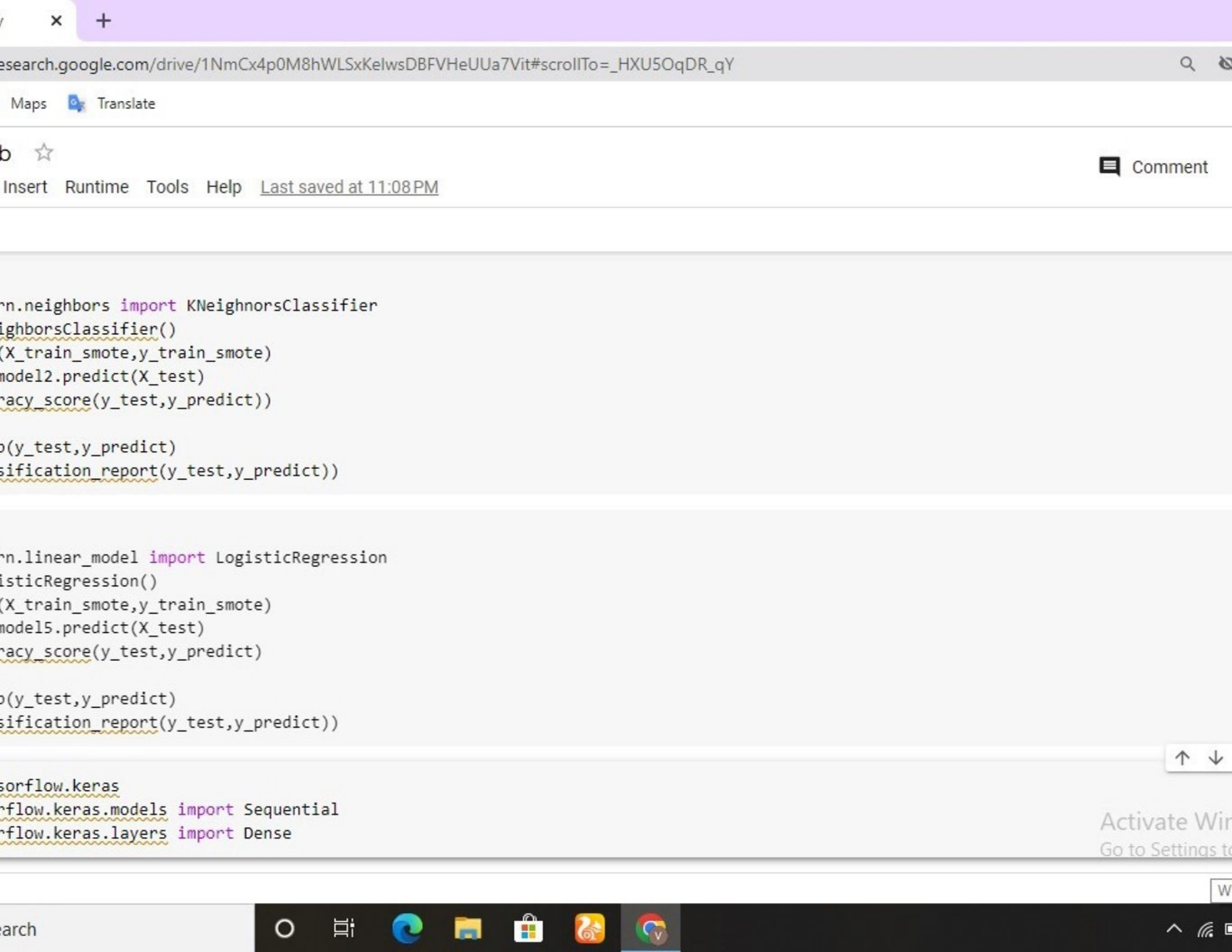


Activate Win
Go to Settings to



research.google.com/drive/1NmCx4p0M8hWLSxKelwsDBFVHeUUa7Vit#scrollTo=_HXU5OqDR_qY

Maps Translate

b ☆

Comment

Insert Runtime Tools Help Last saved at 11:08 PM

```
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier()
knn.fit(X_train_smote, y_train_smote)
y_pred2 = knn.predict(X_test)
accuracy_score(y_test, y_pred2)

confusion_matrix(y_test, y_pred2)
classification_report(y_test, y_pred2)
```

```
from sklearn.linear_model import LogisticRegression
logit = LogisticRegression()
logit.fit(X_train_smote, y_train_smote)
y_pred5 = logit.predict(X_test)
accuracy_score(y_test, y_pred5)

confusion_matrix(y_test, y_pred5)
classification_report(y_test, y_pred5)
```

```
from tensorflow.keras
models import Sequential
layers import Dense
```

Activate Windows
Go to Settings to activate Windows.

```
=Sequential()
```

the input layer and the first hidden layer

```
.add(Dense(units=100,activation='relu',input_dim=10))
```

the second hidden layer

```
.add(Dense(units=50,activation='relu'))
```

the output layer

↑ ↓

the ANN

ANN to the training set

Activate Win
Go to Settings to

```
dict([[50,1,1.2,0.8,150,70,80,7.2,3.4,0.8]])
```

_bilirubin-direct_bilirubin*alkaline_phosphatase-alanin_aminotransferase-aspartate_aminotrans

```
dict([[50,1,1.2,0.8,150,70,80,72,3.4,0.8]])
```

```
.save("liver.hs")
```

```
classifier.predict(X_test)
```

research.google.com/drive/1NmCx4p0M8hWLSxKelwsDBFVHeUUa7Vit#scrollTo=S6nAvu-uXyaS

Maps Translate

b ☆

Comment

Insert Runtime Tools Help Last saved at 11:08 PM

pred>0.5)

exit(sample_value):

to numpy array

ue=np.array(sample_value)

because sample_value contains only 1 record

ue=sample_value.reshape(1,-1)

↑ ↓

aling

Activate Windows
Go to Settings to activate Windows.


```
value=scale(sample_value)
```

```
classifier.predict(sample_value)
```

$$\text{r-total_bilirubin-direct_bilirubin*alkaline_phosphatase}$$

```
ue= [[50,1,1.2,0.8,150,70,80,7.2,3.4,0.8]]
```

```
_exit(sample_value)>0.5:  
prediction:'Liver Patient')
```

```
prediction:'Healthy')
```

Traceback (most recent call last)

```
input-6-e5316e1826e7> in <module>
sample_value=[[50,1,1.2,0.8,150,70,80,7.2,3.4,0.8]]
predict exit(sample_value)>0.5:
```

Activate Win
Go to Settings to


```
from ensemble import ExtraTreesClassifier
et = ExtraTreesClassifier()
et.fit(X,y)
```

```
Frame(model.feature_importances_, index=X.columns).sort_values(0, ascending=False)
```

```
nd='barh',figsize=(7,6))
'FEATURE IMPORTANCE",fontsize=14)
```

```
lib
p(model1, 'ETC.pk1')
```