```
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score,confusion_matrix
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
df = sns.load_dataset('iris')
```

```
df.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

```
encoder = LabelEncoder()
df['species'] = encoder.fit_transform(df['species'])
```

```
df.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0

```
df = df[['sepal_length','petal_length','species']]
```

```
df.head()
```

	sepal_length	petal_length	species
0	5.1	1.4	0
1	4.9	1.4	0
2	4.7	1.3	0
3	4.6	1.5	0
4	5.0	1.4	0

```
X = df.iloc[:,0:2]
y = df.iloc[:,-1]
```

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=2)
```

```
clf = LogisticRegression(multi_class='multinomial')
```

```
clf.fit(X_train,y_train)
```

```
LogisticRegression(multi_class='multinomial')
```

```
y_pred = clf.predict(X_test)
```

```
print(accuracy_score(y_test,y_pred))
```

```
0.9666666666666667
```

```
pd.DataFrame(confusion_matrix(y_test,y_pred))
```

	0	1	2
0	14	0	0
1	0	7	1
2	0	0	8

```
# prediction
query = np.array([[3.4,2.7]])
clf.predict_proba(query)
```

```
array([[7.25957888e-01, 2.73627865e-01, 4.14246954e-04]])
```

```
clf.predict(query)
```

```
array([0])
```

```
from mlxtend.plotting import plot_decision_regions

plot_decision_regions(X.values, y.values, clf, legend=2)

# Adding axes annotations
plt.xlabel('sepal length [cm]')
plt.xlabel('petal length [cm]')
plt.title('Softmax on Iris')

plt.show()
```