

# ass2

October 15, 2018

## 1 Assignment 2

Chiyu Chen

```
In [1]: import pandas as pd
import numpy as np
from sklearn import datasets

In [2]: iris = datasets.load_iris()
iris_df = pd.DataFrame(data= np.c_[iris['data'], iris['target']], columns= iris['feature_names'],
target_names = iris['target_names'])

In [3]: def likelihood(df, sepal_length, sepal_width, petal_length, petal_width):
    likelihood = 1.0
    df_means = df.mean()
    df_std = df.std()
    attr_list = [('sepal length (cm)', sepal_length), ('sepal width (cm)', sepal_width),
    ('petal length (cm)', petal_length), ('petal width (cm)', petal_width)]
    for attrname, attrval in attr_list:
        nominator = np.exp(np.divide(-(np.square(attrval - df_means[attrname])), np.square(df_std[attrname])))
        denominator = np.sqrt(np.pi * 2.0) * df_std[attrname]
        tmp = np.divide(nominator, denominator)
        likelihood *= tmp
    return likelihood

def posterior(df, sepal_length, sepal_width, petal_length, petal_width):
    classes = [0, 1, 2]
    priors = df['target'].value_counts() / df.shape[0]
    posters = []
    for cls in classes:
        pri = priors[cls]
        llh = likelihood(df[df["target"] == cls], sepal_length, sepal_width, petal_length, petal_width)
        posters.append(pri * llh)
    return posters

def max_a_posterior(df, sepal_length, sepal_width, petal_length, petal_width):
    posters = posterior(df, sepal_length, sepal_width, petal_length, petal_width)
    print(posters)
    return np.argmax(posters)
```

```
In [4]: target_names[max_a_posterior(iris_df, 4.5, 3.0, 5.6, 2.1)]  
[2.0931983947119886e-191, 7.233385380727956e-08, 0.0011852765941950457]
```

```
Out[4]: 'virginica'
```

```
In [5]: target_names[max_a_posterior(iris_df, 5.4, 2.6, 4.5, 0.0)]  
[3.937733951289929e-69, 4.276732727915964e-11, 6.010508529199821e-15]
```

```
Out[5]: 'versicolor'
```