李京叡 0616329

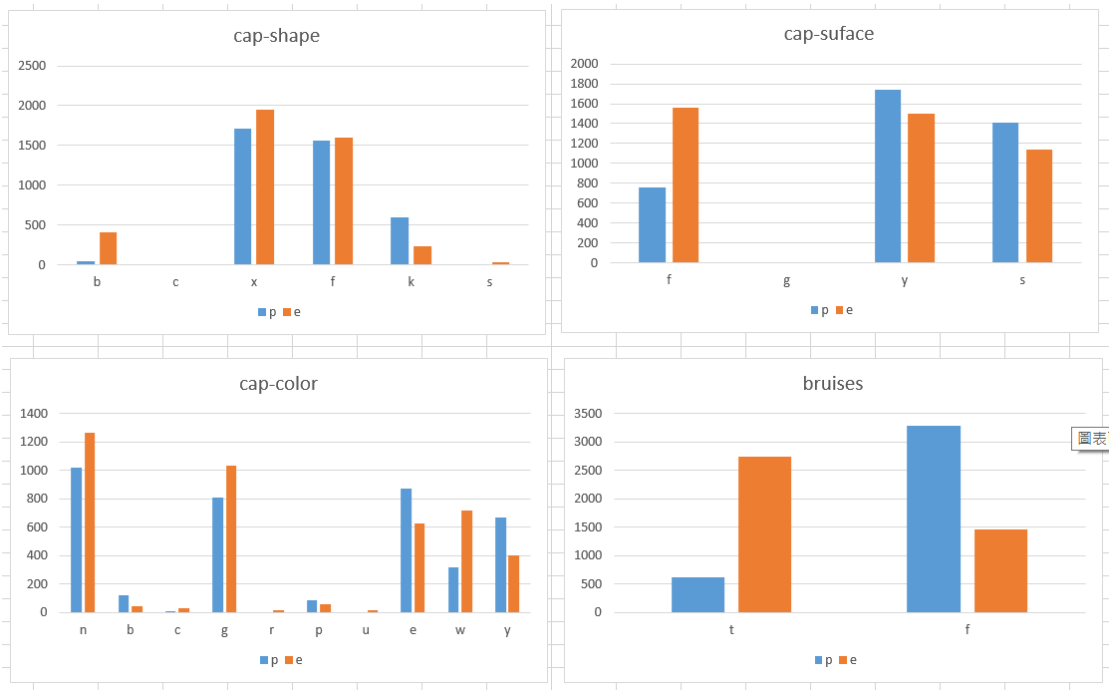
Mushroom

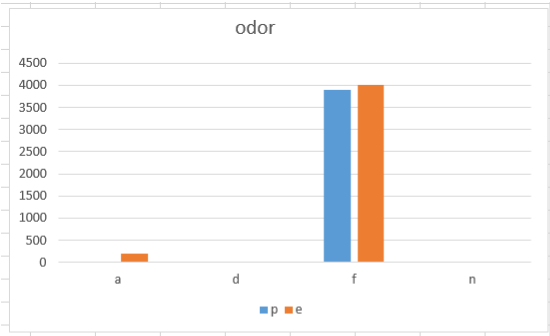
Data input:

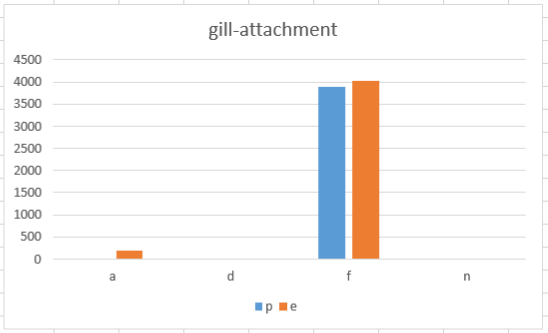
用 pandas 的 read_csv()讀取資料 agaricus-lepiota.data

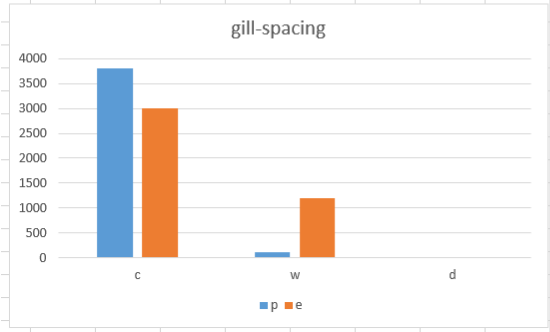
Data Visualization:

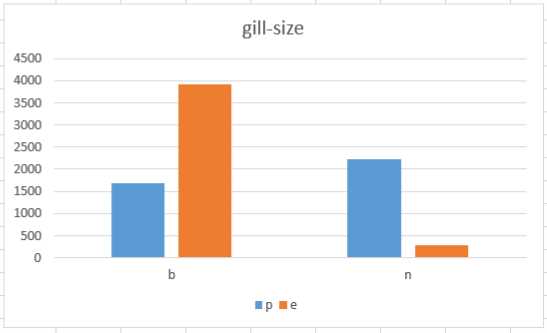
使用 Excel:

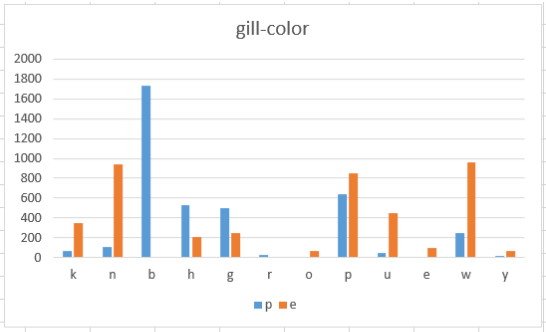


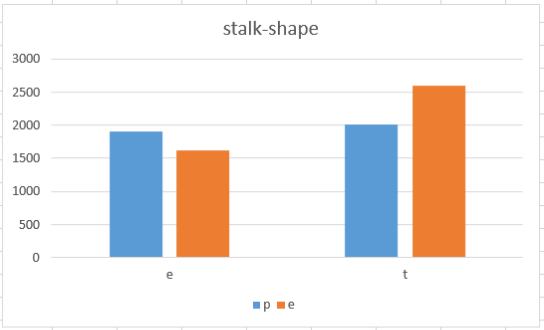

odor
4500
4000
3500
3000
2500
2000
1500
1000
500
0
a
d
f
n
p e

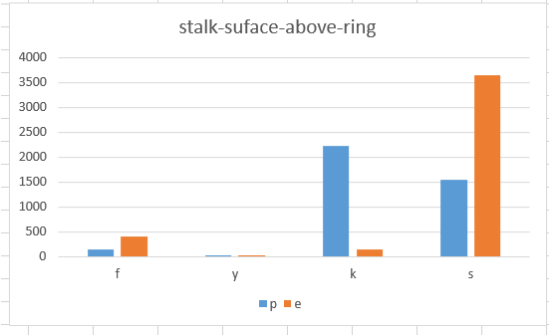

gill-attachment
4500
4000
3500
3000
2500
2000
1500
1000
500
0
a
d
f
n
p e

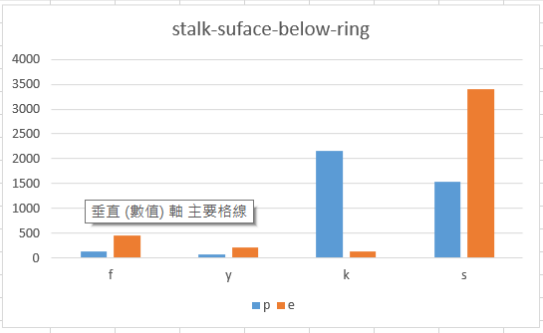

gill-spacing
4000
3500
3000
2500
2000
1500
1000
500
0
c
w
d
p e

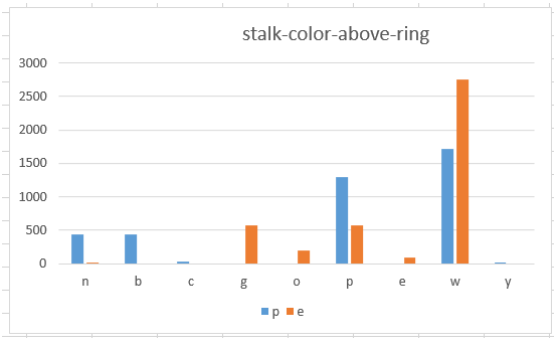

gill-size
4500
4000
3500
3000
2500
2000
1500
1000
500
0
b
n
p e

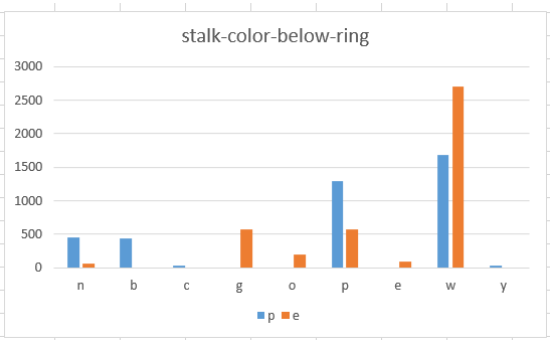

gill-color
2000
1800
1600
1400
1200
1000
800
600
400
200
0
k
n
b
h
g
r
o
p
u
e
w
y
p e

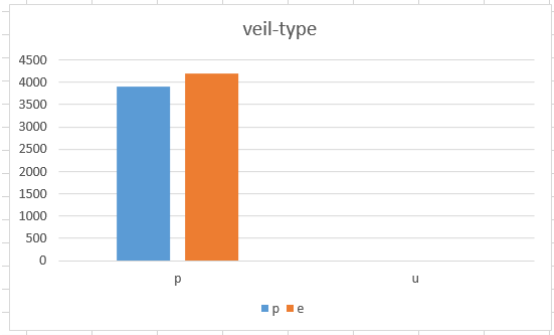

stalk-shape
3000
2500
2000
1500
1000
500
0
e
t
p e

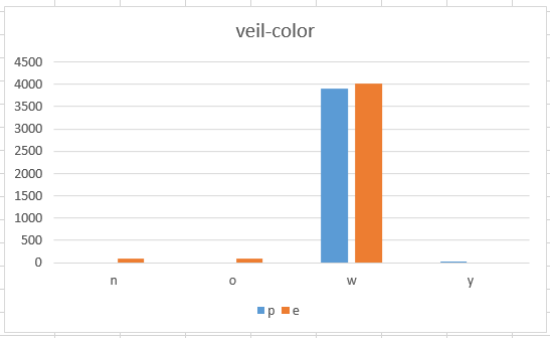

stalk-suface-above-ring
4000
3500
3000
2500
2000
1500
1000
500
0
f
y
k
s
p e

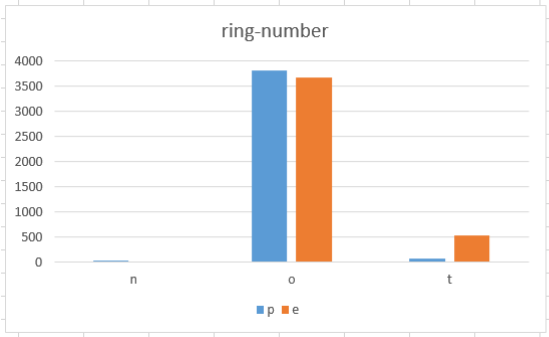

stalk-suface-below-ring
4000
3500
3000
2500
2000
1500
1000
500
0
垂直 (數值) 軸 主要格線
f
y
k
s
p e

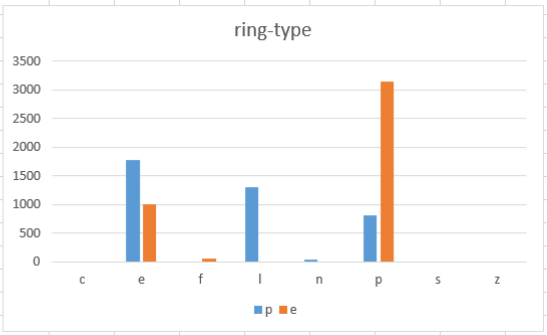


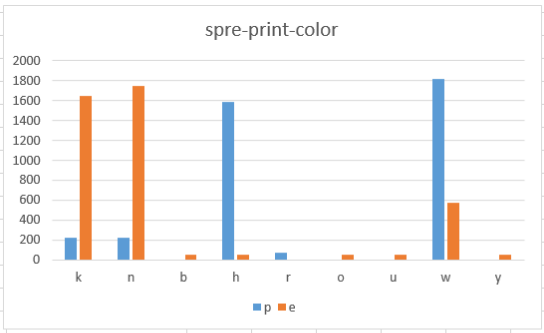


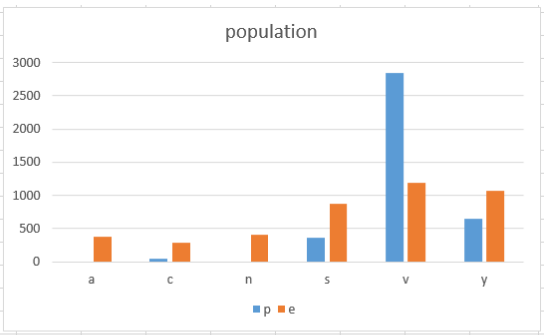


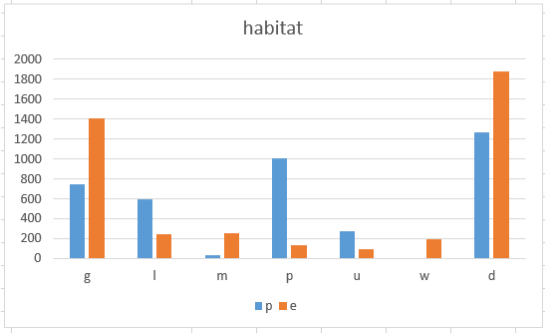












Data preprocessing:

1. 將有 missing 的 feature 移除，作法是先將 ”?” 替換成 nan，再用 dropna()把那一欄刪除
2. 打亂順序

Model construction:

無 Laplace smoothing 的 model: 寫在 naïve_bayes 函式中，主要原理為:

$(\mathbf{q})=\text{argmax}_{Y\in\mathbb{T}}[\log P(Y)+\sum_{mi=1}\log P(X_i|Y)$

$P(Xi|Y)=N(Xi|Y)/N(Y)$

有做 Laplace smoothing 的 model: 寫在 naïve_bayes_smoothing 函式中，原理為:

$(\mathbf{q})=\text{argmax}_{Y\in\mathbb{T}}[\log P(Y)+\sum_{mi=1}\log P(X_i|Y)$

$P(Xi|Y)=(N(Xi|Y)+k)/(N(Y)+k\tau)$，取 k=1

Train-and-spilt:

分為 train : test = 7 : 3

```
train = df.sample(frac=0.7, random_state=200)
test = df.drop(train.index)
```

Results:

```
Result without Laplace smoothing:
                Predicted edible        Predicted poisonous
Actual edible           1158                    123
Actual poisonous        255                     901
Accuracy: 0.844891
Sensitivity: 0.903981
Precision: 0.819533

Result with Laplace smoothing(k=1):
                Predicted edible        Predicted poisonous
Actual edible           1277                    4
Actual poisonous        97                      1059
Accuracy: 0.958556
Sensitivity: 0.996877
Precision: 0.929403
```

Without Laplace smoothing:

Confusion Matrix:

	Predicted edible	Predicted poisonous
Actual edible	1158	123
Actual poisonous	255	901

Accuracy: 0.884891

Sensitivity: 0.903981

Precision: 0.819533

With Laplace smoothing:

Confusion Matrix

	Predicted edible	Predicted poisonous
Actual edible	1277	4
Actual poisonous	97	1059

Accuracy: 0.958556

Sensitivity: 0.996877

Precision: 0.929403

Comparison & Conclusion:

因為每次都是取隨機的樣本來訓練 model，所以結果每次都不一樣，但是有做 Laplace smoothing 的 model 都會有較高的 Accuracy, Sensitivity, Precision，所以顯示做 Laplace smoothing 可使 model 準確度提高。

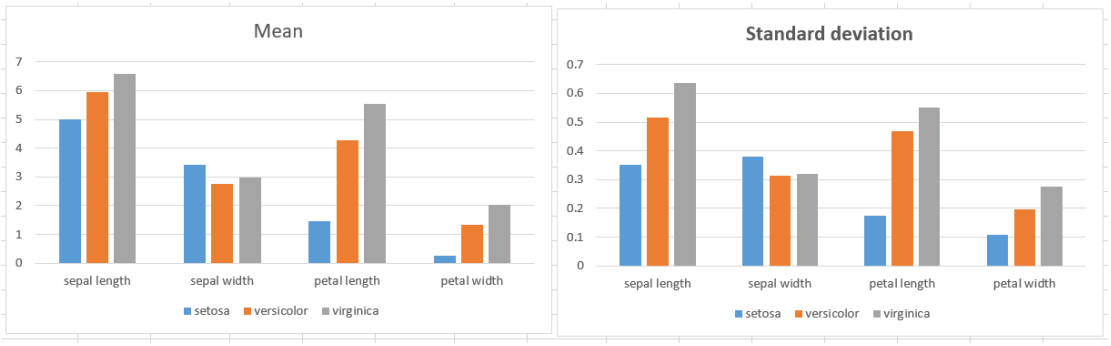
Iris

Data input:

用 pandas 的 read_csv()讀取資料 iris.data

Data Visualization:

使用 Excel:



Data preprocessing:

打亂順序

Model construction:

$(\mathbf{q})=\text{argmax}Y\in\mathbb{T}[\log P(Y)+\sum mi=1\log P(Xi|Y)$

$(Xi|Y)=1/(\sigma\sqrt{2\pi})\exp(-(x-\mu)^2/(2\sigma^2))$

Train-and-spilt:

分為 train : test = 7 : 3

```
train = df.sample(frac=0.7, random_state=200)
test = df.drop(train.index)
```

Results:

	Predicted Setosa	Predicted Virginica	Predicted Versicolour
Actual Setosa	14	0	0
Actual Virginica	0	18	1
Actual Versicolour	0	12	0

Confusion Matrix:

	Predicted Setosa	Predicted Virginica	Predicted Versicolour
Actual Setosa	14	0	0
Actual Virginica	0	18	1
Actual Versicolour	0	12	0

Accuracy: 0.711111

Precision (Setosa = True): 1

Sensitivity (Setosa = True): 1

Precision (Virginica = True): 0.947368

Sensitivity (Virginica = True): 0.6

Precision (Versicolour = True): 0

Sensitivity (Versicolour = Ture): 0

Comparison & Conclusion:

在這個 model 中 Versicolour 會被誤認為是 Virginica，可能是因為 Versicolour 這個品種的花和 Virginica 沒有太突出的特徵。