

$$(X, O) = e^{-\frac{x^2}{2\sigma^2}}$$

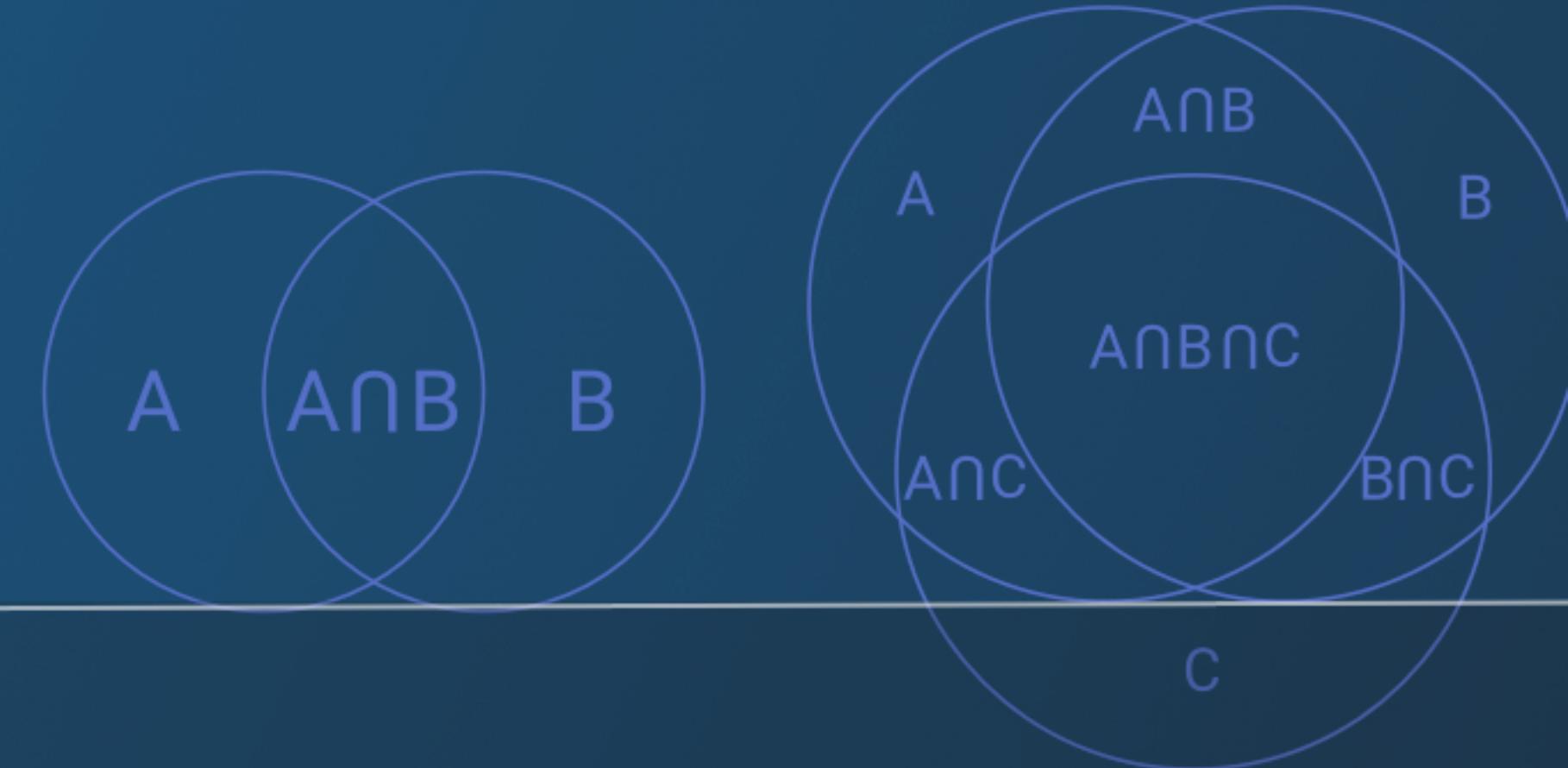
$$x(X, O) = -\frac{x}{\sigma^2} G(X, O) = -\frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}}$$

$$xx(X, O) = \frac{x^2 - \sigma^2}{\sigma^4} G(X, O) = \frac{x^2 - \sigma^2}{\sigma^4} e^{-\frac{x^2}{2\sigma^2}}$$

$$xxx(X, O) = -\frac{x^3 - x\sigma^2}{\sigma^6} G(X, O) = -\frac{x^3 - x\sigma^2}{\sigma^6} e^{-\frac{x^2}{2\sigma^2}}$$

$$= x \ln(x + \sqrt{1 + x^2})$$

Julia 程式語言學習馬拉松



Day 30



cupay 陪跑專家 : Andy Tu

DecisionTree 決策樹





重要知識點



- 認識 DecisionTree 套件
- 了解決策樹、隨機森林與 Adaboost tree 模型
- 了解模型的訓練及預測方式



DecisionTree 套件



- DecisionTree 套件提供了決策樹、隨機森林與 Adaboost tree 模型。
- 隨機森林這類整體學習模型常常是競賽排行榜上的前幾名，是效果很好的模型。
- 這類模型可以適用於輸入資料有連續數值及類別型資料混合的情況。



準備資料

- 這邊以 iris 資料集做示範。

```
data = RDatasets.dataset("datasets",
"iris")
features = Matrix(data[1:4])
labels = Vector[String](data[:Species])
```



模型訓練

- 決策樹模型有分類模型及迴歸模型，訓練是使用 `fit!` 函式。
- 其參數眾多，多數與樹的修剪相關，可以參考官方手冊以及資料結構樹的相關知識。

```
model =  
DecisionTree.DecisionTreeClassifier(max_ de  
pth=2)  
DecisionTree.fit!(model, features, labels)
```



模型預測

- 模型預測的使用非常簡單，使用 predict 函式。

```
new_iris = [5.9, 3.0, 5.1, 1.9]
DecisionTree.predict(model, new_iris)
```



整體學習

- 整體學習 (ensemble learning) 是將許多弱模型加以組合成為一個強大模型的技術。
- 它可以被廣泛適用於眾多模型上，大大增強模型的性能。
- 隨機森林與 Adaboost tree 則是其中的代表，分別使用了 bagging 及 boosting 的技術。



隨機森林模型

- 隨機森林模型有分類模型及迴歸模型。
- 隨機森林模型的參數與決策樹的雷同。
- 訓練及預測方式也相同。

```
model =  
DecisionTree.RandomForestClassifier()
```



模型種類



分類

迴歸

決策樹

DecisionTreeClassifier

DecisionTreeRegressor

隨機森林

RandomForestClassifier

RandomForestRegressor

Adaboost

AdaBoostStumpClassifier

-



Cross validation

- 在模型評估部份，由於套件本身並沒有提供，目前仰賴 ScikitLearn 套件的交叉驗證功能。
- 或是想要使用其他套件提供的功能也可以，也歡迎大家自己撰寫評估的函式，這樣可以更進一步了解評估的方法。

```
using ScikitLearn.CrossValidation:  
cross_val_score  
accuracy = cross_val_score(model,  
features, labels, cv=5)
```

知識點 回顧

- 認識 DecisionTree 套件
- 了解決策樹、隨機森林與 Adaboost tree 模型
- 了解模型的訓練及預測方式



推薦閱讀

- [DecisionTree.jl 官方文件](#)





解題時間

請跳出 PDF 至官網 Sample Code
& 作業開始解題