ScikitLearn 操作記錄單 2

組別: Team16　　學號: 41071102H　　姓名: 徐敏皓

Supervised Learning

1. 請根據以下教學資源操作: http://www.cse.msu.edu/~ptan/dmbook/tutorials/tutorial6/tutorial6.html
2. 請自行查詢了解下列 scikit-learn 模組的功能作用 https://scikit-learn.org/stable/

Classification	Module	Function	試寫程式，實驗該函式所提供功能及主要參數設定效果
K-Neighbors classification	sklearn.neighbors	KNighborsClassifier ()	功能: 基於 k 最近鄰 (k-Nearest Neighbors, KNN) 算法的分類模型。它的核心功能是根據輸入的特徵，找出在訓練資料中最接近的 k 個鄰居，並根據這些鄰居的類別來預測新資料的類別。 KNighborsClassifier (n_neighbors, weights, algorithm, leaf_size, p, metric, metric_params, n_jobs) n_neighbors: 指定使用多少個最近的鄰居。預設為 5。

			weights: 預設為 'uniform'。 • 'uniform': 所有鄰居的權重相等。 • 'distance': 鄰居的權重與距離成反比。 • callable: 自定義函數來計算權重。 algorithm: 用於計算最近鄰的演算法。預設為'auto'。 • 'auto': 根據數據自動選擇合適的演算法。 • 'ball_tree': 使用 Ball Tree。 • 'kd_tree': 使用 KD Tree。 • 'brute': 強行計算距離。 leaf_size: 用於 Ball Tree 或 KD Tree 的葉節點大小，影響查找速度與內存使用。預設為30。

			p: 距離的冪次參數。當 p=1 使用曼哈頓距離，p=2 使用歐幾里得距離。預設為 2。 metric: 計算距離的方法。預設為’ minkowski’ • 'euclidean' • 'manhattan' • 'minkowski' metric_params: 用於距離計算的額外參數。預設為 None。 n_jobs: 使用 CPU 的核心數。-1 代表使用所有可用的核心。預設為 None。
	sklearn.neighbors	KNighborsRegressor ()	功能: 基於 k 最近鄰 (K-Nearest Neighbors, KNN) 演算法的回歸模型，通常用於解決回歸問題（預測數值型目標變數）。 KNighborsRegressor (n_neighbors, weights, algorithm, leaf_size, p, metric, metric_params, n_jobs)

			參數設定效果與 KNighborsClassifier 一樣。
Naïve Bayes Classifiers	sklearn.naive_bayes	Gaussian Naive Bayes()	功能: 基於高斯分佈的單純貝氏分類器，主要用於處理連續型數據。 GaussianNB(priors, var_smoothing) priors: 類別的先驗機率。如果指定了該參數，模型將使用給定的先驗機率，而不從數據中估算。 var_ smoothing: 用於數值穩定性的平滑參數，將最大特徵變異數的一小部分加入到每個特徵的變異數中，以避免出現零值導致的除零錯誤。預設為 1e-9。
	sklearn.naive_bayes	MultinomialNB()	功能: 用於分類問題，特別適用於特徵呈現為計數或頻率數據的情況，例如文本分類。 MultinomialNB(alpha, fit_prior, class_prior,

			normalize_prior) alpha: 平滑參數，用於防止零概率問題（Laplace smoothing）。alpha 越小，平滑效應越弱，越接近原始的最大似然估計。預設為 1 fit_prior: 是否學習類別的先驗概率。如果設為 False，則每個類別的先驗概率將等同於類別在訓練數據中的頻率。預設為 True。 class_prior: 如果不為 None，則它是一個長度為 n_classes 的列表，表示每個類別的先驗概率。預設為 None。 normalize_prior: 是否正規化先驗概率。若設為 False，則會使用 class_prior 直接作為先驗。預設為 True。

Decision Trees Classification	sklearn.tree	DecisionTreeClassifier()	功能: 用於分類任務的機器學習模型，屬於決策樹（Decision Tree）算法的一種實現。 DecisionTreeClassifier(criterion, max_depth, min_samples_split, min_samples_leaf, max_features, random_state, max_leaf_nodes, min_impurity_decrease, class_weight) criterion: 用來決定劃分的標準。預設為'gini'。 • "gini": 使用 Gini impurity（基尼不純度）。 • "entropy": 使用信息增益（information gain）。 max_depth: 決定樹的最大深度。如果設為None，則樹會擴展直到所有葉子節點都是純粹的，或者包含少於 min_samples_split 的樣本。預設為 None。

			min_samples_split: 每個內部節點再劃分所需的最小樣本數。可以是整數（表示樣本數），也可以是浮動數（表示比例）。預設為 2。 min_samples_leaf: 每個葉子節點所需的最小樣本數。預設為 1。 max_features: 在劃分時要考慮的最大特徵數。可以是整數、浮動數（比例）、"auto"、"sqrt" 或 "log2"。預設為 None。 random_state: 用來控制隨機性。如果為整數，則控制隨機種子。預設為 None。 max_leaf_nodes: 設定葉子節點的最大數量。如果為 None，則無限制。預設為 None。 min_impurity_decrease: 需要減少的最小不純

			度，才會進行劃分。預設為 0。 class_weight: 類別權重，可以用來平衡不同類別的樣本數。預設為 None。
	sklearn.tree	DecisionTreeRegressor()	功能: 通過決策樹來預測連續數值目標變量（即回歸問題） DecisionTreeRegressor(criterion, splitter, max_depth, min_samples_split, min_samples_leaf, max_features, random_state, max_leaf_nodes, min_impurity_decrease) criterion: 決定樹的分割標準。 • 預設值："squared_error"（均方誤差） • 可選值："absolute_error"（絕對誤差）或其他根據任務適合的損失函數。 splitter: 分割策略。 • 預設值："best"（選擇局部最佳分割）

			• 可選值："random"（隨機分割） 其餘參數設定效果與 DecisionTreeClassifier 相同。
SVM Classification	Sklearn.svm	LinearSVC()	功能: 用於解決二元或多類別分類問題。 LinearSVC(penalty, loss, dual, tol, C, multi_class, max_iter, class_weight, random_state) penalty: 懲罰項類型，預設值為 'l2'。可以設為 'l1' 或 'l2'。 loss: 損失函數，預設值為 'squared_hinge'。可以設為 'hinge' 或 'squared_hinge'。 dual: 是否求解對偶問題，預設值為 True。如果樣本數量較大或特徵數量小，設為 False。 tol: 收斂容忍度，預設值為 1e-4。 C: 正則化參數，預設值為 1.0。較小的值會增

			加正則化強度。 multi_class: 多類別分類策略，預設值為 'ovr' (one-vs-rest)。可以設為 'crammer_singer'。 max_iter: 最大迭代次數，預設值為 1000。 class_weight: 類別權重，可設為 'balanced' 或字典形式。 random_state: 隨機種子，影響模型穩定性。
	Sklearn.svm	SVC()	功能: 一個最佳的超平面，將不同類別的數據點分開。 SVC(C, kernel, degree, gamma, coef0, probability, class_weight, max_iter, random_state) C: 正則化參數（default=1.0）。控制錯誤的懲罰力度，值越小，模型越簡單。 kernel: 核函數類型（default='rbf'）。常見選

			項： • 'linear'：線性核 • 'poly'：多項式核 • 'rbf'：徑向基函數核（默認） • 'sigmoid'：S 型核 degree: 多項式核函數的次數（只在 kernel='poly' 時有用）。 gamma: 核係數（default='scale'）。影響 RBF、poly 和 sigmoid 核函數，值可以是： • 'scale'：1 / (特徵數 * 標準差) • 'auto'：1 / 特徵數 coef0: 核函數中的獨立項（只對 poly 和 sigmoid 核有用）。

			probability: 是否啟用概率估計 （default=False）。 class_weight: 用於處理不平衡數據的類別權 重。 max_iter: 最大迭代次數（default=-1，表示無 限制）。 random_state: 隨機數種子（用於概率估計）。
ANN Classification	Sklearn. neural_network	MLPClassifier()	功能: 基於前向傳播的神經網絡，適合解決非 線性分類問題。 MLPClassifier(hidden_layer_sizes, activation, solver, alpha, learning_rate, learning_rate_init, max_iter, random_state, verbose, early_stopping) hidden_layer_sizes: 指定隱藏層的神經元數量， 例如 (100,) 表示一層有 100 個神經元，(100, 50) 表示兩層，分別有 100 和 50 個神經元。

			activation: 激活函數，可選： • 'identity': 線性函數 • 'logistic': Sigmoid 函數 • 'tanh': 雙曲正切 • 'relu': 修正線性單元 (預設) solver: 權重優化算法： • 'lbfgs': 梯度優化 • 'sgd': 隨機梯度下降 • 'adam': 自適應動量估計 (預設) alpha: L2 正則化的懲罰項係數，默認為 0.0001。 learning_rate: 學習率調整方法： • 'constant': 固定學習率

			• 'invscaling': 逐漸減小學習率 • 'adaptive': 當 loss 停止改善時減少學習率 learning_rate_init: 初始學習率，默認為 0.001。 max_iter: 最大迭代次數，默認為 200。 random_state: 隨機數種子，用於重現結果。 verbose: 是否輸出訓練過程的日誌。 early_stopping: 是否啟用早停，如果驗證集的效果停止改善，則提前停止。
	Sklearn. neural_network	MLPRegressor()	功能: 通過前向傳播和反向傳播學習輸入特徵與目標變數之間的關係。用於解決非線性回歸問題。 MLPRegressor (hidden_layer_sizes, activation,

			solver, alpha, learning_rate, learning_rate_init, max_iter, random_state, verbose, tol) tol: 停止訓練的容忍度，預設為 1e-4。 其餘參數設定效果與 MLPClassifier 相同。
Ensemble classifier	Sklearn.ensemble	RandomForestClassifier ()	功能: 通過建立多棵決策樹，並將它們的預測結果進行投票，來進行分類。 RandomForestClassifier(n_estimators, criterion, max_depth, min_samples_split, min_samples_leaf, max_features, random_state, n_jobs) n_estimators: 預設值: 100。隨機森林中樹的數量，增加樹的數量可以提高模型性能，但計算時間也會增加。 criterion: 預設值: 'gini'。衡量分割質量的標準，選項包括 'gini'和 'entropy'。 max_depth: 預設值: None。樹的最大深度，設置此參數可以防止過擬合。

			min_samples_split: 預設值: 2。分割內部節點所需的最小樣本數。 min_samples_leaf: 預設值: 1。葉子節點所需的最小樣本數。 max_features: 預設值: 'sqrt'。用於最佳分割的特徵數目，可選值包括 'sqrt', 'log2', 或 None。 random_state: 預設值: None。設置隨機種子以確保結果可重現。 n_jobs: 預設值: None。使用多少個 CPU 核心並行運算。設為 -1 時使用所有可用的核心。
	Sklearn.ensemble	GradientBoostingClassifier() GradientBoostingRegressor()	1. 功能: 通過將多個弱分類器（通常是決策樹）組合成一個強分類器，以提高預測的準確性。 GradientBoostingClassifier(n_estimators,

			learning_rate, max_depth, min_samples_split, min_samples_leaf, subsample, max_features, loss) n_estimators: 設定弱學習器（決策樹）的數量，預設值是 100。 learning_rate: 用於控制每棵樹對最終預測的貢獻，預設值是 0.1。較小的值通常會提高模型的穩定性，但需要更多的樹來達到相同的效果。 max_depth: 決策樹的最大深度，預設值是 3。深度越大，模型的複雜度也越高。 min_samples_split: 內部節點再分裂所需的最小樣本數，預設值是 2。 min_samples_leaf: 葉子節點上最小的樣本數量，預設值是 1。

			subsample: 用於控制每棵樹所用的樣本比例，範圍是 (0.0, 1.0)，預設是 1.0（使用全部樣本）。 max_features: 用來控制每棵樹隨機選擇的特徵數量，預設是 None（每棵樹使用所有特徵）。 loss: 損失函數，常見的有 deviance（對應對數損失）和 exponential（對應指數損失），預設是 deviance。 ---------- 2. 功能: 通過結合多個弱學習器（如迴歸樹），逐步優化模型性能。 GradientBoostingRegressor(loss, learning_rate, n_estimators, subsample, criterion, max_depth, min_samples_split, min_samples_leaf, max_features, random_state, validation_fraction, n_iter_no_change)

			loss: 指定損失函數，可選 "squared_error"（均方誤差），"absolute_error"（絕對值誤差），"huber"，"quantile"。 criterion: 分裂節點時的標準（默認 "friedman_mse"）。 random_state: 控制隨機性（默認 None）。 validation_fraction: 用於早停的驗證集比例。 n_iter_no_change: 早停時容許的最長無改進迭代次數。 其餘參數設定效果與 GradientBoostingClassifier 相同。
Evaluation	Sklearn.model_selection	KFold()	功能: 評估機器學習模型在不同數據分割上的性能，提升結果的可靠性。 KFold(n_splits, shuffle, random_state)

			n_splits: 指定將數據分成幾個折疊。必填參數，默認值為 5。 shuffle: 是否在分割前隨機打亂數據。默認值為 False。 random_state: 如果 shuffle=True，用於設置隨機種子以確保可重現性。默認值為 None。
	Sklearn.model_selection	ShuffleSplit()	功能: 交叉驗證方法，用於將數據集隨機分割成訓練集和測試集。 ShuffleSplit(n_splits, test_size, train_size, random_state) n_splits: 訓練/測試分割的次數，表示會生成多少次不同的訓練集和測試集。 test_size: 測試集的比例（0.0 到 1.0 之間）或測試集的數量（如果是 int）。如果為 None，

			則會自動選擇一個合理的比例。 train_size: 訓練集的比例（0.0 到 1.0 之間）或訓練集的數量（如果是 int）。如果是 None，則會由 test_size 自動推算。 random_state: 用於隨機數生成的種子，這樣每次分割結果會一致。如果是 None，則會基於當前時間進行隨機分割。
	Sklearn.metrics	confusion_matrix() classification_report() f1_score() precision_recall_curve()	1. 功能: 評估分類模型的性能，呈現實際值與預測值的對比，特別適用於二元或多類別分類問題。 confusion_matrix(y_true, y_pred, labels, sample_weight, normalize) y_true: 真實標籤 (1D array-like) y_pred: 預測標籤 (1D array-like)

			labels: 指定標籤順序 (list-like)，若未提供，將自動推斷。 sample_weight: 樣本權重 (array-like) normalize: 是否將混淆矩陣標準化 ('true', 'pred', 'all' 或 None，默認為 None) --- 2. 功能: 評估分類模型表現的函數。它會計算並輸出多種常見的分類指標，包括 Precision, Recall, F1-score 等。 classification_report(y_true, y_pred, labels, target_names, sample_weight, digits, output_dict, zero_division) y_true: 真實的標籤，通常是測試集的標籤。 y_pred: 預測的標籤，通常是模型對測試集的預測結果。 labels: 需要計算報告的標籤類別。如果設置為

			None，則會自動使用 y_true 和 y_pred 中的所有標籤。預設為 None。 target_names: 顯示每個類別的名稱。這是用來代替類別的數字標籤來顯示的名稱。預設為 None。 sample_weight: 每個樣本的權重，會影響計算指標的結果。預設為 None。 digits: 報告中顯示的小數位數。預設為 2。 output_dict: 是否以字典的形式返回結果。如果為 True，將返回包含各項指標的字典，否則返回字符串格式的報告。預設為 False。 zero_division: 預設為'warn'。用來處理除以零的情況，通常出現在某個類別的預測完全為 0

			時。選項有： • 'warn'：顯示警告 • 0：返回 0 • 1：返回 1 --- 3. 功能: 評估分類模型性能的指標，用於衡量模型對正類（Positive Class）的精確率（Precision）與召回率（Recall）之間的平衡。 f1_score(y_true, y_pred, labels, pos_label, average, zero_division) y_true: 實際標籤（Ground truth）。 y_pred: 預測標籤（Predicted labels）。 labels (可選): 設定用來計算分數的標籤子集。 pos_label (可選): 二元分類時，指定「正類」的標籤，默認為 1。

			average (可選): • 'binary': 用於二元分類（默認）。 • 'micro': 計算全局的 TP、FP 和 FN。 • 'macro': 計算每個類別的 F1 分數，然後取平均，不考慮類別不平衡。 • 'weighted': 根據類別支持度（support）計算加權平均。 • 'samples': 多標籤分類時使用。 zero_division (可選): 當除以零時，返回 0 或 1，或者拋出警告（"warn"，默認）。 --- 4. 功能: 用於計算 精確率 (Precision) 和 召回率 (Recall) 的函數，並生成一條以不同閾值劃分的 Precision-Recall 曲線。其功能主要是評

			估 二元分類模型 的性能，尤其在 不平衡數據集 中非常有用。 precision_recall_curve(y_true, probas_pred, pos_label) y_true: 真實標籤（實際值），為一維數組或列表，數值為二元分類的類別標籤（通常為 0 或 1）。 probas_pred: 預測的分數或機率，為模型輸出的置信分數或概率值。 pos_label: 指定正類的標籤，預設值為 1。

補充(regression model)

http://www.cse.msu.edu/~ptan/dmbook/tutorials/tutorial5/tutorial5.html

其他參考資源:

- machine learning 參考書: "Introduction to Machine Learning with Python" 之 github code

https://github.com/amueller/introduction_to_ml_with_python/blob/master/02-supervised-learning.ipynb
https://github.com/amueller/introduction_to_ml_with_python/blob/master/05-model-evaluation-and-improvement.ipynb

Scikit Learn documentation(http://scikit-learn.org/stable/index.html)

- 尋搜尋其他可信網路資源