

# קובץ ניסויים – מטלה 1

אציג את התוצאות של כל ניסוי בצורה הבאה:

עבור כל מסווג אציג ראשית את התוצאות שמתקבלות עם הערכים הדיפולטיביים, ולאחר מכן, עבור כל שינוי אציג בעזרת תמונה את הערכים שנתתי לכל פרמטר, ומיד לאחר מכן אציג בעזרת תמונה את התוצאות של כל ניסוי.

## 1. עבור המסווג הראשון - Decision Tree Classifier :

### הניסוי הראשון – הערכים הדיפולטיביים

הערכים הדיפולטיביים הם :

```
model_decision_tree = DecisionTreeClassifier(criterion='gini', splitter='best', max_depth=None,
                                             min_samples_split=2, min_samples_leaf=1,
                                             min_weight_fraction_leaf=0.0, max_features=None,
                                             random_state=None, max_leaf_nodes=None, min_impurity_decrease=0.0,
                                             class_weight=None, ccp_alpha=0.0, monotonic_cst=None)
```

והתוצאות עבורם הם :

Decision Tree Results:

Accuracy: 0.9123

Precision: 0.9154

Recall: 0.9123

F1-Score: 0.9127

Confusion Matrix:

```
[[44  3]
 [ 7 60]]
```

### הניסוי השני :

כעת, נשנה קצת את הערכים ובמקום gini ננסה entropy ונגדיר גובה מקסימלי 6 ונראה את התוצאות. אז, הערכים שלי עכשיו הם :

```
(criterion='entropy', splitter='best', max_depth=6,
 min_samples_split=2, min_samples_leaf=1,
 min_weight_fraction_leaf=0.0, max_features=None,
 random_state=None, max_leaf_nodes=None, min_impurity_decrease=0.0,
 class_weight=None, ccp_alpha=0.0, monotonic_cst=None)
```

ונראה שהתוצאות השתפרו טיפה (וזה ממש משמח כי זה מגניב שהשפענו על התוצאות בשינוי של 2 פרמטרים סה"כ)

## והתוצאות הם:

```
Decision Tree Results:  
Accuracy: 0.9211  
Precision: 0.9230  
Recall: 0.9211  
F1-Score: 0.9214  
Confusion Matrix:  
[[44  3]  
 [ 6 61]]
```

## הניסוי השלישי:

עכשיו ננסה במקום entropy, נחליף ל log\_loss ונשנה את העומק המקסימלי ל 8, ואת הפיצול המינימלי ל 4:

```
(criterion='log_loss', splitter='best', max_depth=8,  
min_samples_split=4, min_samples_leaf=1,  
min_weight_fraction_leaf=0.0, max_features=None,  
random_state=None, max_leaf_nodes=None, min_impurity_decrease=0.0,  
class_weight=None, ccp_alpha=0.0, monotonic_cst=None)
```

ונקבל את התוצאות היותר טובות אפילו מהניסוי הקודם:

```
Decision Tree Results:  
Accuracy: 0.9386  
Precision: 0.9404  
Recall: 0.9386  
F1-Score: 0.9388  
Confusion Matrix:  
[[45  2]  
 [ 5 62]]
```

## הניסוי הרביעי:

עכשיו אנסה להחזיר ל gini ונשנה את החלוקה המינימלית ל 6, את העומק ל 10, ונתן ערך רנדומי של 5 (במקום NONE)

```
(criterion='gini', splitter='best', max_depth=10,  
min_samples_split=6, min_samples_leaf=1,  
min_weight_fraction_leaf=0.0, max_features=None,  
random_state=5, max_leaf_nodes=None, min_impurity_decrease=0.0,  
class_weight=None, ccp_alpha=0.0, monotonic_cst=None)
```

ועכשיו הרסנו קצת כי התוצאות פחות טובות :

```
Decision Tree Results:  
Accuracy: 0.9123  
Precision: 0.9154  
Recall: 0.9123  
F1-Score: 0.9127  
Confusion Matrix:  
[[44  3]  
 [ 7 60]]
```

### הניסוי החמישי :

ננסה לשנות ל entropy , וננסה לשנות הפעם את splitter ל רנדומלי העומק המקסימלי ל 15 , את החלוקה המינימלית ל 4 , ואת הערך הרנדומי ל 66 ,

```
(criterion='entropy', splitter='random', max_depth=15,  
min_samples_split=4, min_samples_leaf=1,  
min_weight_fraction_leaf=0.0, max_features=None,  
random_state=66, max_leaf_nodes=None, min_impurity_decrease=0.0,  
class_weight=None, ccp_alpha=0.0, monotonic_cst=None)
```

ונקבל את התוצאות הכי טובות עד כה שהן:

```
Decision Tree Results:  
Accuracy: 0.9561  
Precision: 0.9604  
Recall: 0.9561  
F1-Score: 0.9564  
Confusion Matrix:  
[[47  0]  
 [ 5 62]]
```

## הניסוי השישי :

ננסה לחזור ל gini , אבל נשנה את העומק המקסימלי ל 6 ,  
את החלוקה המינימלית ל 4 , ואת הערך הרנדומי ל 55:

```
(criterion='gini', splitter='best', max_depth=6,  
min_samples_split=4, min_samples_leaf=1,  
min_weight_fraction_leaf=0.0, max_features=None,  
random_state=55, max_leaf_nodes=None, min_impurity_decrease=0.0,  
class_weight=None, ccp_alpha=0.0, monotonic_cst=None)
```

ונקבל את התוצאות:

Decision Tree Results:

Accuracy: 0.9474

Precision: 0.9474

Recall: 0.9474

F1-Score: 0.9474

Confusion Matrix:

```
[[44  3]  
 [ 3 64]]
```

ולכן , עבור המסווג הראשון , נשאיר בקוד את הניסיון החמישי  
שלי שיצא עם התוצאות הכי טובות.

## 2. עבור המסווג השני - Random Forest Classifier :

### הניסוי הראשון – הערכים הדיפולטיביים

הערכים הדיפולטיביים הם :

```
model_random_forest = RandomForestClassifier(n_estimators=100, criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1,  
min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None, min_impurity_decrease=0.0,  
bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False,  
class_weight=None, ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

Random Forest Results:

Accuracy: 0.9561

Precision: 0.9578

Recall: 0.9561

F1-Score: 0.9563

Confusion Matrix:

```
[[46  1]
```

והתוצאות עבורם הם:

## הניסוי השני:

נשנה קצת את הערכים ונבדוק מה קורה כשמשנים את העומק המקסימלי ל 8, את הפיצול המינימלי ל 4, ואת הערך הרנדומי ל 55.  
אז הערכים:

```
(n_estimators=100, criterion='gini', max_depth=8, min_samples_split=4, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=55, verbose=0, warm_start=False, class_weight=None, ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

והתוצאות לא השתנו כמעט, אבל עדיין טיפה פחות טובות :

Random Forest Results:

Accuracy: 0.9561

Precision: 0.9564

Recall: 0.9561

F1-Score: 0.9562

Confusion Matrix:

```
[[45  2]
 [ 3 64]]
```

## הניסוי השלישי:

עכשיו ננסה לשנות את gini ל entropy, את העומק המקסימלי ל 12, את הפיצול המינימלי ל 6, ואת הערך הרנדומי ל 42 :

```
(n_estimators=100, criterion='entropy', max_depth=12, min_samples_split=6, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=42, verbose=0, warm_start=False, class_weight=None, ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

ושיפרנו יפה את התוצאות :

Random Forest Results:

Accuracy: 0.9649

Precision: 0.9649

Recall: 0.9649

F1-Score: 0.9649

Confusion Matrix:

```
[[45  2]
 [ 2 65]]
```

## הניסוי הרביעי:

ננסה לשנות כך שנחזיר לערך הרנדומי NONE, נשנה את העומק ל 2, ונשנה את הפיצול המינימלי ל 10, ומספר הדוגמאות המקסימלי ל 15 ( האמת שעשיתי את זה בכוונה כדי לראות שכשאני נותן מספר דוגמאות קטן יחסית זה פוגע בתוצאות כמו שאני מצפה שיקרה )  
הפרמטרים:

```
(n_estimators=100, criterion='log_loss', max_depth=2, min_samples_split=10, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=20, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, class_weight=None, ccp_alpha=0.0, max_samples=15, monotonic_cst=None)
```

ואכן, פגענו בתוצאות שיצאו הפעם :

```
Random Forest Results:  
Accuracy: 0.9386  
Precision: 0.9385  
Recall: 0.9386  
F1-Score: 0.9385  
Confusion Matrix:  
[[43  4]  
 [ 3 64]]
```

## הניסוי החמישי:

כעת נשנה את העומק המקסימלי ל 6, פיצול מינימלי ל 4, גם את min\_samples\_leaf ל 4, וגם את max\_leaf\_node ל 20, וערך רנדומי 45.  
אז הערכים:

```
(n_estimators=100, criterion='log_loss', max_depth=6, min_samples_split=4, min_samples_leaf=4, min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=20, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=45, verbose=0, warm_start=False, class_weight=None, ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

והתוצאות יצאו הכי טובות שקיבלנו עד כה :

```
Random Forest Results:  
Accuracy: 0.9737  
Precision: 0.9738  
Recall: 0.9737  
F1-Score: 0.9736  
Confusion Matrix:  
[[45  2]  
 [ 1 66]]
```

## הניסוי השישי:

נשנה את העומק המקסימלי ל 12, פיצול מינימלי ל 4, גם את min\_samples\_leaf ל 8, וגם את max\_leaf\_node ל 30.

אז הערכים:

```
(n_estimators=100, criterion='log_loss', max_depth=12, min_samples_split=4, min_samples_leaf=8, min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=30, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, class_weight=None, ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

Random Forest Results:                   : והתוצאות

```
Accuracy: 0.9649
Precision: 0.9657
Recall: 0.9649
F1-Score: 0.9650
Confusion Matrix:
[[46  1]
 [ 3 64]]
```

ולכן, שוב הניסיון החמישי הוא הכי טוב ולכן נשאיר אותו.

## 3. עבור המסווג השלישי - Ada Boost Classifier :

## הניסוי הראשון – הערכים הדיפולטיביים

הערכים הדיפולטיביים הם :

```
model_adaboost = AdaBoostClassifier(estimator=None, n_estimators=50,
                                     learning_rate=1.0, algorithm='SAMME.R', random_state=None)
```

והתוצאות לערכים הדיפולטיביים:

```
AdaBoost Results:
Accuracy: 0.9561
Precision: 0.9561
Recall: 0.9561
F1-Score: 0.9561
Confusion Matrix:
[[44  3]
 [ 2 65]]
```

## הניסוי השני:

נשנה את הפרמטרים `n_estimators` ל 100 , דירוג למידה ל 1.5 וערך רנדומי ל 55 :

```
(estimator=None, n_estimators=100,  
learning_rate=1.5, algorithm='SAMME.R', random_state=55)
```

ונראה ששיפרנו את התוצאות :

```
AdaBoost Results:  
Accuracy: 0.9737  
Precision: 0.9739  
Recall: 0.9737  
F1-Score: 0.9737  
Confusion Matrix:  
[[46  1]  
 [ 2 65]]
```

## הניסוי השלישי:

נשנה את הערך הרנדומי ל 42, את דירוג הלמידה ל 2.0 ואת `n_estimators` ל 250:

```
(estimator=None, n_estimators=250,  
learning_rate=2.0, algorithm='SAMME.R', random_state=42)
```

ונראה שפגענו משמעותית בתוצאות:

```
AdaBoost Results:  
Accuracy: 0.9123  
Precision: 0.9133  
Recall: 0.9123  
F1-Score: 0.9116  
Confusion Matrix:  
[[40  7]  
 [ 3 64]]
```

## הניסוי הרביעי:

נחזיר את הערך הרנדומי ל NONE, את דירוג הלמידה נשנה ל 0.7 ואת `n_estimators` ל 80:



```
(estimator=None, n_estimators=80,  
learning_rate=0.7, algorithm='SAMME.R', random_state=None)
```

והתוצאות:

```
AdaBoost Results:  
Accuracy: 0.9649  
Precision: 0.9649  
Recall: 0.9649  
F1-Score: 0.9649  
Confusion Matrix:  
[[45  2]  
 [ 2 65]]
```

### הניסוי החמישי:

נשנה את הערך הרנדומי ל 88, את דירוג הלמידה ל 3.25 ואת  
ה n\_estimators ל 185:

```
(estimator=None, n_estimators=185,  
learning_rate=3.25, algorithm='SAMME.R', random_state=88)
```

והתוצאות:

```
AdaBoost Results:  
Accuracy: 0.8421  
Precision: 0.8755  
Recall: 0.8421  
F1-Score: 0.8328  
Confusion Matrix:  
[[29 18]  
 [ 0 67]]
```

### הניסוי השישי:

נשנה את הערך הרנדומי ל 60, את דירוג הלמידה ל 2.5 ואת  
ה n\_estimators ל 177:

```
(estimator=None, n_estimators=177,  
learning_rate=2.5, algorithm='SAMME.R', random_state=60)
```

והתוצאות:

```
AdaBoost Results:  
Accuracy: 0.8421  
Precision: 0.8755  
Recall: 0.8421  
F1-Score: 0.8328  
Confusion Matrix:  
[[29 18]  
 [ 0 67]]
```

אז, הניסוי הכי טוב במסווג השלישי הוא הניסוי השני ולכן נקח את התוצאות שלו.

שני הניסויים עם התוצאות הכי טובות, קרובים מאוד בתוצאות, והם:

הניסוי השני של Ada Boost Classifier

והניסוי החמישי של Random Forest Classifier

בין שני הניסויים האלה, הניסוי של Ada Boost Classifier יותר טוב ויש שני מדדים שהם שוברים שיוויון f-sorce וגם precision, ואני אבחר ב f-sorce כמדד המכריע.

ולכן Ada Boost Classifier ניצח ( ציינתי כמובן גם בקוד )