ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
Τμήμα Τεχνολογιών Ψηφιακής Βιομηχανίας

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Εφαρμογή Μηχανικής Μάθησης και Βαθιάς Μάθησης στη Διάγνωση Κινητήρων και Αναγνώριση Οδικών Σημάτων

Πέτρος Γερόσταθος
Α.Μ.: 1117201900025

Επιβλέπων: Γεώργιος Αλεξανδρίδης, Επίκουρος Καθηγητής

ΑΘΗΝΑ

ΜΑΪΟΣ 2024

ABSTRACT

This thesis focuses on the application of machine learning and deep learning techniques to two different problems: engine condition prediction and traffic sign recognition.

1. **Engine Condition Prediction using MLP and SVM:**Data from engine sensors were used to predict its condition.Preprocessing involved category balancing through oversampling and normalization using standard scaling. Two different models were tested: a Multi-Layer Perceptron (MLP) and a Support Vector Machine (SVM).The models' hyperparameters were optimized using grid search and cross-validation, achieving improved accuracy in predicting the engine condition.
2. **Convolutional Neural Network (CNN)** was trained to classify 12 different types of traffic signs.Image preprocessing included conversion to grayscale, lighting normalization, and data normalization to a 0-1 scale.The CNN was designed with multiple convolutional and fully connected layers, and dropout was applied to improve generalization.
3. **Model Evaluation for Traffic Signs:**The trained CNN was evaluated on an independent test set.The overall accuracy of the model and the individual success rates for each type of sign were recorded, providing insights into its performance. This thesis demonstrates that the application of these techniques is effective for prediction and classification problems, highlighting the advantages of machine learning and deep learning in industrial and everyday scenarios.

SUBJECT AREA: Usage of Machine Learning and Artificial Intelligence for Diagnostics and Recognition in the Automotive Industry

KEYWORDS: Machine Learning, Artificial Intelligence, Car Engines, Road Signs, Problem Diagnosis, Preventive Maintenance, Automotive Industry

ΠΕΡΙΛΗΨΗ

Η παρούσα εργασία ασχολείται με την εφαρμογή τεχνικών μηχανικής μάθησης και βαθιάς μάθησης για δύο διαφορετικά προβλήματα. Την πρόβλεψη της κατάστασης ενός κινητήρα και την αναγνώριση οδικών σημάτων.

1. **Πρόβλεψη κατάστασης κινητήρα με MLP και SVM:** Χρησιμοποιήθηκαν δεδομένα από κινητήρα για να προβλεφθεί η κατάστασή του. Μετά από επεξεργασία που περιλάμβανε την εξισορρόπηση των κατηγοριών μέσω oversampling και κανονικοποίηση με standard scaling, δοκιμάστηκαν δύο διαφορετικά μοντέλα: ένα Πολυεπίπεδο Νευρωνικό Δίκτυο (MLP) και ένας Υποστηρικτικός Διανυσματικός Μηχανισμός (SVM).Οι υπερπαράμετροι των μοντέλων βελτιστοποιήθηκαν μέσω grid search και cross-validation, επιτυγχάνοντας βελτιωμένη ακρίβεια στην πρόβλεψη της κατάστασης του κινητήρα.

2. **Αναγνώριση οδικών σημάτων με CNN:** Εκπαιδεύτηκε ένα Convolutional Neural Network (CNN) για την ταξινόμηση 12 διαφορετικών τύπων οδικών σημάτων. Προηγήθηκε προεπεξεργασία εικόνων, περιλαμβάνοντας μετατροπή σε ασπρόμαυρη, εξίσωση φωτισμού, και κανονικοποίηση των δεδομένων σε 0-1. Το CNN σχεδιάστηκε με πολλαπλά επίπεδα συνελικτικών και πλήρως συνδεδεμένων στρωμάτων, ενώ εφαρμόστηκε dropout για τη βελτίωση της γενίκευσης.

3. **Αξιολόγηση του μοντέλου για οδικά σήματα:**Το εκπαιδευμένο CNN αξιολογήθηκε σε ένα ανεξάρτητο σύνολο δοκιμής. Η συνολική ακρίβεια του μοντέλου και τα επιμέρους ποσοστά επιτυχίας για κάθε τύπο σήματος καταγράφηκαν, παρέχοντας πληροφορίες για την απόδοσή του. Η εργασία αποδεικνύει πως η εφαρμογή αυτών των τεχνικών είναι αποτελεσματική σε προβλήματα πρόβλεψης και ταξινόμησης, αναδεικνύοντας τα πλεονεκτήματα της μηχανικής μάθησης και βαθιάς μάθησης σε βιομηχανικά και καθημερινά σενάρια.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Χρήση Μηχανικής Μάθησης και Τεχνητής Νοημοσύνης για Διαγνωστική και Αναγνώριση στην Αυτοκινητοβιομηχανία

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Μηχανική Μάθηση, Τεχνητή Νοημοσύνη, Κινητήρες Αυτοκινήτων, Οδικά Σήματα, Διάγνωση Προβλημάτων, Πρόληψη Συντήρησης, Αυτοκινητοβιομηχανία

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες σε όλους εκείνους που συνέβαλαν στην ολοκλήρωση της διπλωματικής μου εργασίας. Πρώτα απ’ όλα, ευχαριστώ από καρδιάς τον επιβλέποντα καθηγητή μου, κ. Γεώργιο Αλεξανδρίδη, για την πολύτιμη καθοδήγηση, την υποστήριξη και την εμπιστοσύνη που μου έδειξε καθ' όλη τη διάρκεια της ερευνητικής μου προσπάθειας.

Η συμβολή του ήταν καθοριστική για την επιτυχή ολοκλήρωση της εργασίας μου. Θα ήθελα επίσης να ευχαριστήσω τον Υποψήφιο Διδάκτορα Καρποντίνη Δημήτρη για τη σημαντική του συνεισφορά στο ερευνητικό μέρος της εργασίας μου.

Επιπλέον, θα ήθελα να ευχαριστήσω και τον κ. Διονύση Ξενάκη για την έμπνευση και την καθοδήγηση που μου έδωσε στα ακαδημαϊκά μου χρόνια και που ήταν πάντα διαθέσιμος και πρόθυμος να με βοηθήσει σε κάθε δυσκολία.

Επίσης, ευχαριστώ όλους τους καθηγητές μου για τις γνώσεις που μου προσέφεραν κατά τη διάρκεια των σπουδών μου. Η αφοσίωσή τους στην εκπαίδευση και η προθυμία τους να με βοηθήσουν σε κάθε μου απορία, ήταν πηγή έμπνευσης και υποστήριξης καθ' όλη την ακαδημαϊκή μου πορεία.

CONTENTS

LIST OF FIGURES

1. ΕΙΣΑΓΩΓΗ

Η παρούσα πτυχιακή εργασία επικεντρώνεται στην εφαρμογή τεχνικών μηχανικής μάθησης και βαθιάς μάθησης για δύο κύρια προβλήματα: την πρόβλεψη της κατάστασης ενός κινητήρα και την αναγνώριση οδικών σημάτων. Στόχος της εργασίας είναι η ανάπτυξη αποτελεσματικών μοντέλων που μπορούν να αξιοποιηθούν για τη βελτίωση της απόδοσης και της ασφάλειας σε βιομηχανικά και καθημερινά σενάρια.

Η πρόβλεψη της κατάστασης του κινητήρα πραγματοποιήθηκε χρησιμοποιώντας διάφορες τεχνικές προεπεξεργασίας και μηχανικής μάθησης. Τα δεδομένα κινητήρα φορτώθηκαν από ένα σύνολο δεδομένων και χωρίστηκαν σε χαρακτηριστικά και στόχους. Η αρχική προεπεξεργασία περιλάμβανε την εξισορρόπηση των κατηγοριών μέσω υπερδειγματοληψίας και την τυποποίηση των δεδομένων μέσω standard scaling.

Στη συνέχεια, εφαρμόστηκαν δύο μοντέλα μηχανικής μάθησης για την πρόβλεψη της κατάστασης του κινητήρα: το Πολυεπίπεδο Νευρωνικό Δίκτυο (MLP) και ο Υποστηρικτικός Διανυσματικός Μηχανισμός (SVM). Τα μοντέλα αυτά δοκιμάστηκαν με τις αρχικές τους παραμέτρους και στη συνέχεια έγινε βελτιστοποίηση των υπερπαραμέτρων τους μέσω GridSearchCV για την επίτευξη καλύτερης ακρίβειας.

Για την αναγνώριση οδικών σημάτων, αναπτύχθηκε ένα συνελικτικό νευρωνικό δίκτυο (CNN). Οι εικόνες των οδικών σημάτων προεπεξεργάστηκαν μέσω μετατροπής τους σε ασπρόμαυρες και εξισορρόπησης του φωτισμού. Το CNN σχεδιάστηκε με πολλαπλά επίπεδα συνελίξεων και πλήρως συνδεδεμένων στρωμάτων, με την εφαρμογή dropout για τη βελτίωση της γενίκευσης του μοντέλου. Το μοντέλο εκπαιδεύτηκε σε δεδομένα εικόνας που αντιπροσωπεύουν 12 διαφορετικές κατηγορίες οδικών σημάτων. Τα δεδομένα χωρίστηκαν σε εκπαιδευτικό, επικυρωτικό και δοκιμαστικό σύνολο, και το CNN εκπαιδεύτηκε για 15 εποχές.

1.1 Στόχοι πτυχιακής

Πρόβλεψη Κατάστασης Κινητήρα: Χρήση δεδομένων κινητήρα για την πρόβλεψη της κατάστασής του μέσω μοντέλων μηχανικής μάθησης.

Αναγνώριση Οδικών Σημάτων: Ανάπτυξη και εκπαίδευση ενός συνελικτικού νευρωνικού δικτύου για την αναγνώριση διαφορετικών τύπων οδικών σημάτων.

1.2 Εργαλεία που χρησιμοποιήθηκαν

1. **Python:** Η κύρια γλώσσα προγραμματισμού για την ανάλυση δεδομένων και την ανάπτυξη μοντέλων.
2. **Pandas και NumPy:** Για τη διαχείριση και την ανάλυση δεδομένων.
3. **Scikit-learn:** Για την ανάπτυξη και την αξιολόγηση μοντέλων μηχανικής μάθησης.
4. **Keras και TensorFlow:** Για την ανάπτυξη και την εκπαίδευση συνελικτικών νευρωνικών δικτύων.
5. **Seaborn και Matplotlib:** Για την απεικόνιση δεδομένων και αποτελεσμάτων

1.3 Περιγραφή των datasets

1.3.1 engine.dataset

Το παρόν dataset περιλαμβάνει δεδομένα σχετικά με την κατάσταση και την απόδοση κινητήρων. Αποτελείται από 19.535 γραμμές και περιέχει τις ακόλουθες στήλες:

1. **Engine rpm:**Στροφές κινητήρα ανά λεπτό (rpm). Τύπος δεδομένων: int64.
2. **Lub oil pressure:**Πίεση λιπαντικού ελαίου σε bar. Τύπος δεδομένων: float64.
3. **Fuel pressure:**Πίεση καυσίμου σε bar. Τύπος δεδομένων: float64.
4. **Coolant pressure:**Πίεση ψυκτικού υγρού σε bar. Τύπος δεδομένων: float64.
5. **Lub oil temp:** Θερμοκρασία λιπαντικού ελαίου σε °C. Τύπος δεδομένων: float64.
6. **Coolant temp:**Θερμοκρασία ψυκτικού υγρού σε °C. Τύπος δεδομένων: float64.
7. **Engine Condition:**Κατάσταση κινητήρα (0: μη καλή, 1: καλή). Τύπος δεδομένων: int64.

1.3.2 Στατιστική Περίληψη

- **Engine rpm:** Η μέση τιμή είναι 791.2 rpm με τυπική απόκλιση 267.6. Οι τιμές κυμαίνονται από 61 έως 2239 rpm.
- **Lub oil pressure:** Η μέση τιμή είναι 3.3 bar με τυπική απόκλιση 1.0. Οι τιμές κυμαίνονται από 0.003 έως 7.27 bar.
- **Fuel pressure:** Η μέση τιμή είναι 6.66 bar με τυπική απόκλιση 2.76. Οι τιμές κυμαίνονται από 0.003 έως 21.14 bar.
- **Coolant pressure:**Η μέση τιμή είναι 2.34 bar με τυπική απόκλιση 1.04. Οι τιμές κυμαίνονται από 0.002 έως 7.48 bar.
- **Lub oil temp:** Η μέση τιμή είναι 77.64 °C με τυπική απόκλιση 3.11. Οι τιμές κυμαίνονται από 71.32 έως 89.58 °C.
- **Coolant temp:**Η μέση τιμή είναι 78.43 °C με τυπική απόκλιση 6.21. Οι τιμές κυμαίνονται από 61.67 έως 195.53 °C.
- **Engine Condition:**Η μέση τιμή είναι 0.63, με τυπική απόκλιση 0.48. Οι τιμές είναι είτε 0 είτε 1.

1.3.3 Δείγμα Δεδομένων

Engine rpm	Lub oil pressure	Fuel pressure	Coolant pressure	Lub oil temp	Coolant temp	Engine Condition
700	2.493592	11.790927	3.178981	84.144163	81.632187	1
876	2.941606	16.193866	2.464504	77.640934	82.445724	0
520	2.961746	6.553147	1.064347	77.752266	79.645777	1
473	3.707835	19.510172	3.727455	74.129907	71.774629	1
619	5.672919	15.738871	2.052251	78.396989	87.000225	0

Σχήμα 1.1: Δείγμα Δεδομένων

Αυτό το dataset μπορεί να χρησιμοποιηθεί για την ανάλυση της απόδοσης των κινητήρων, την πρόβλεψη της κατάστασης τους με βάση τις διάφορες μετρήσεις και την ανάπτυξη αλγορίθμων συντήρησης.

1.3.4 Traffic sign recognition

Το German Traffic Sign Benchmark είναι ένα dataset που χρησιμοποιείται για την ταξινόμηση σημάτων κυκλοφορίας. Το συγκεκριμένο dataset δημιουργήθηκε και παρουσιάστηκε στο Διεθνές Συνέδριο για Νευρωνικά Δίκτυα (IJCNN) το 2011 και έχει τα ακόλουθα χαρακτηριστικά:

1. **Τύπος Προβλήματος:**Πρόκειται για ένα πρόβλημα μονοεικονικής, πολυκλασικής ταξινόμησης. Ο όρος "πολυκλασική" αναφέρεται στο γεγονός ότι το πρόβλημα περιλαμβάνει πολλές διαφορετικές κατηγορίες. Στην περίπτωση της αναγνώρισης σημάτων κυκλοφορίας, οι κατηγορίες αυτές μπορεί να περιλαμβάνουν διάφορους τύπους σημάτων ενώ Ο όρος "μονοεικονική" αναφέρεται στο γεγονός ότι το μοντέλο μας λαμβάνει μία εικόνα κάθε φορά ως είσοδο. Κάθε εικόνα που εισάγεται στο σύστημα περιέχει ένα μόνο σήμα κυκλοφορίας.
2. **Αριθμός Κλάσεων:**Το dataset περιλαμβάνει 12 κλάσεις σημάτων κυκλοφορίας μετα την επεξεργασια του.
3. **Αριθμός Εικόνων:**Συνολικά περιέχει πάνω από 15.000 εικόνες..
4. **Βάση Δεδομένων:** Η βάση δεδομένων είναι μεγάλη και ρεαλιστική, παρέχοντας ένα πλούσιο σύνολο δεδομένων για την εκπαίδευση και αξιολόγηση αλγορίθμων ταξινόμησης.

1.3.5 Δομή του Dataset

Το dataset περιλαμβάνει εικόνες σημάτων κυκλοφορίας , οι οποίες είναι οργανωμένες σε φακέλους ανά κλάση. Κάθε φάκελος περιέχει εικόνες ενός συγκεκριμένου σήματος κυκλοφορίας. Υπάρχει επίσης ένα αρχείο ετικετών που συσχετίζει κάθε εικόνα με την αντίστοιχη κλάση της. Ιδιότητες των Εικόνων

- **Μέγεθος:**Οι εικόνες μπορεί να έχουν ποικίλα μεγέθη, αλλά συχνά κλιμακώνονται σε σταθερές διαστάσεις (π.χ., 32x32 ή 64x64 pixels).
- **Μορφή:**Οι εικόνες συνήθως είναι σε μορφή PNG ή JPEG.
- **Χρώμα:**Οι εικόνες είναι έγχρωμες (RGB).

Το dataset αυτό είναι εξαιρετικά χρήσιμο για την ανάπτυξη και αξιολόγηση αλγορίθμων μηχανικής μάθησης και τεχνητής νοημοσύνης, ειδικά στον τομέα της αναγνώρισης σημάτων κυκλοφορίας. Η μεγάλη ποικιλία των κλάσεων και ο ρεαλιστικός χαρακτήρας των εικόνων το καθιστούν ένα πολύτιμο εργαλείο.

Σχήμα 1.2: Όριο ταχύτητας 100

Σχήμα 1.3: Σήμα STOP

Σχήμα 1.4: Σήμα απαγόρευσης

2. ΒΗΜΑΤΑ ΠΡΟΕΠΕΞΕΡΓΑΣΙΑΣ

2.1 Α) Για τον κινητήρα :

1. Αρχικά εισάγουμαι τις απαραίτητες βιβλιοθήκες:
2. **pandas** (συχνά αναφερόμενο ως **pd**) για χειρισμό δεδομένων.
3. **numpy** (συχνά αναφερόμενο ως **np**) για αριθμητικές λειτουργίες.
4. **seaborn** (συχνά αναφερόμενο ως **sns**) για τη δημιουργία γραφημάτων.
5. **matplotlib.pyplot** (συχνά αναφερόμενο ως **plt**) για τη δημιουργία γραφημάτων.

2.2 Φόρτωση δεδομένων:

Χρησιμοποιείται η συνάρτηση **pd.read_csv()** για να φορτώσει τα δεδομένα μας από το αρχείο **CSV engine_data.csv**.

Διαχωρισμός χαρακτηριστικών και στόχου:

- **target** (στόχος) είναι η στήλη **'Engine Condition'** από τα δεδομένα.
- **features** (χαρακτηριστικά) είναι όλα τα υπόλοιπα δεδομένα χωρίς τη στήλη **'Engine Condition'**.
- Εκτύπωση των πρώτων πέντε γραμμών των χαρακτηριστικών: Χρησιμοποιείται η συνάρτηση **head()** για να εμφανίσει τις πρώτες πέντε γραμμές των χαρακτηριστικών.
- Εκτύπωση των πρώτων πέντε γραμμών του στόχου: Χρησιμοποιείται η συνάρτηση **target.head()** για να εμφανίσει τις πρώτες πέντε γραμμές του στόχου.
- Στην συνεχεια επαναλαμβανουμε την παραπανω διαδικασια για να ξανα δουμε την μορφη του χωρισμενου **engine_data** αυτήν την φορα.
- Χρησιμοποιείται η συνάρτηση **features.head()** αυτην την φορα για να εμφανίσει τις πρώτες πέντε γραμμές των χαρακτηριστικών. Εμφανίζει τις πρώτες 5 γραμμές του **DataFrame features**

	Engine rpm	Lub oil pressure	Fuel pressure	Coolant pressure	lub oil temp	Coolant temp
0	700	2.493592	11.790927	3.178981	84.144163	81.632187
1	876	2.941606	16.193866	2.464504	77.640934	82.445724
2	520	2.961746	6.553147	1.064347	77.752266	79.645777
3	473	3.707835	19.510172	3.727455	74.129907	71.774629
4	619	5.672919	15.738871	2.052251	78.396989	87.000225

Σχήμα 2.1: 5 ΠΡΩΤΕΣ ΓΡΑΜΜΕΣ DATAFRAME

	Engine rpm	Lub oil pressure	Fuel pressure	Coolant pressure	lub oil temp	Coolant temp
19530	902	4.117296	4.981360	4.346564	75.951627	87.925087
19531	694	4.817720	10.866701	6.186689	75.281430	74.928459
19532	684	2.673344	4.927376	1.903572	76.844940	86.337345
19533	696	3.094163	8.291816	1.221729	77.179693	73.624396
19534	504	3.775246	3.962480	2.038647	75.564313	80.421421

Σχήμα 2.2: 5 ΤΕΛΕΥΤΑΙΕΣ ΓΡΑΜΜΕΣ DATAFRAME

Έπειτα με την χρήση της συνάρτησης **features.tail()** θα εμφανιστούν οι τελευταίες 5 γραμμες των χαρακτιρηστηκών.

Για να δούμε ολοκληρωμένα το σχήμα του θα χρησιμοποιήσουμε την συνάρτηση **features.shape()** το οποίο μας εμφανίζει το σχήμα του **DataFrame features**, το οποίο είναι (19535,6). Αυτό σημαίνει ότι το **dataset** έχει 19535 γραμμές και 6 στήλες.

Με την εντολή **features.describe()** κάνουμε μια στατιστική περίληψη του **dataset** η οποια εντολη μας εμφανίζει στατιστικά περιγραφικά στοιχεία για κάθε στήλη του **DataFrame**, όπως το μέσο όρο, τη διακύμανση, το ελάχιστο και το μέγιστο.

	Engine rpm	Lub oil pressure	Fuel pressure	Coolant pressure	lub oil temp	Coolant temp
count	19535.000000	19535.000000	19535.000000	19535.000000	19535.000000	19535.000000
mean	791.239263	3.303775	6.655615	2.335369	77.643420	78.427433
std	267.611193	1.021643	2.761021	1.036382	3.110984	6.206749
min	61.000000	0.003384	0.003187	0.002483	71.321974	61.673325
25%	593.000000	2.518815	4.916886	1.600466	75.725990	73.895421
50%	746.000000	3.162035	6.201720	2.166883	76.817350	78.346662
75%	934.000000	4.055272	7.744973	2.848840	78.071691	82.915411
max	2239.000000	7.265566	21.138326	7.478505	89.580796	195.527912

Σχήμα 2.3: Describe dataset

Από το παραπάνω σχεδιάγραμμα συμπεραίνουμε ότι ο κώδικας σημειώνει ότι η διακύμανση των χαρακτηριστικών, όπως μετριέται από την τυπική απόκλιση, είναι ικανοποιητική. Ως εκ τούτου, σε αυτή τη φάση, δεν χρησιμοποιείται το **VarianceThreshold**, το οποίο είναι μια τεχνική για την αφαίρεση χαρακτηριστικών με χαμηλή διακύμανση.

Εμφανίζουμε τα ονόματα των στηλών του **DataFrame features** με την εντολή **features.columns** και έπειτα με την εντολή **features.nunique()** η οποία μας εμφανίζει τον αριθμό των μοναδικών τιμών για κάθε στήλη. Για παράδειγμα, η στήλη **'Engine rpm'** έχει 1379 μοναδικές τιμές, ενώ οι άλλες στήλες έχουν σχεδόν όλες μοναδικές τιμές. Για να εμφανίσουμε τις μοναδικές τιμές μιας στήλης χρησιμοποιούμαι την εντολή **features['Fuel pressure'].unique()** εμφανίζοντας μας τις μοναδικές τιμές της στήλης **'Fuel pressure'**.

```
Engine rpm           1379
Lub oil pressure    19534
Fuel pressure       19531
Coolant pressure    19534
lub oil temp        19530
Coolant temp        19532
dtype: int64
```

Σχήμα 2.4: Features unique

Με αυτές τις εντολές γίνεται η αρχική εξερεύνηση του **dataset features** για να κατανοήσει τη δομή του, τις βασικές στατιστικές ιδιότητες και τις μοναδικές τιμές κάθε στήλης. Αυτή η ανάλυση βοηθά στην κατανόηση των δεδομένων και στην προετοιμασία για περαιτέρω επεξεργασία και ανάλυση.

2.3 Επισκόπηση, καθαρισμός dataset :

1. Ξεκινάμε με την εντολή **features.isnull().sum()** που εμφανίζει τον αριθμό των κενών τιμών σε κάθε στήλη. Εδώ, όλες οι τιμές είναι μηδενικές, που σημαίνει ότι δεν υπάρχουν κενές τιμές στο **dataset**.

```
Engine rpm           0
Lub oil pressure     0
Fuel pressure        0
Coolant pressure     0
lub oil temp         0
Coolant temp         0
dtype: int64
```

Σχήμα 2.5: Features null

2. Ξανά εμφανίζουμε τις πρώτες 5 γραμμές του **dataset** με **features.head()** και έπειτα δημιουργούμαι και εμφανίζουμε διαγράμματα **boxplot** ορίζοντας μια συνάρτηση για τη δημιουργία των **boxplots** για συγκεκριμένες στήλες του **DataFrame**. Στη συνέχεια καλούμαι τη συνάρτηση αυτή για να δημιουργήσει **boxplots** για τις στήλες που αναφέρονται στη λίστα **columns _to_ plot**. Με αυτόν τον τρόπο οπτικοποιούμαι τις στήλες μας για να δούμε τις ακραίες τιμές.

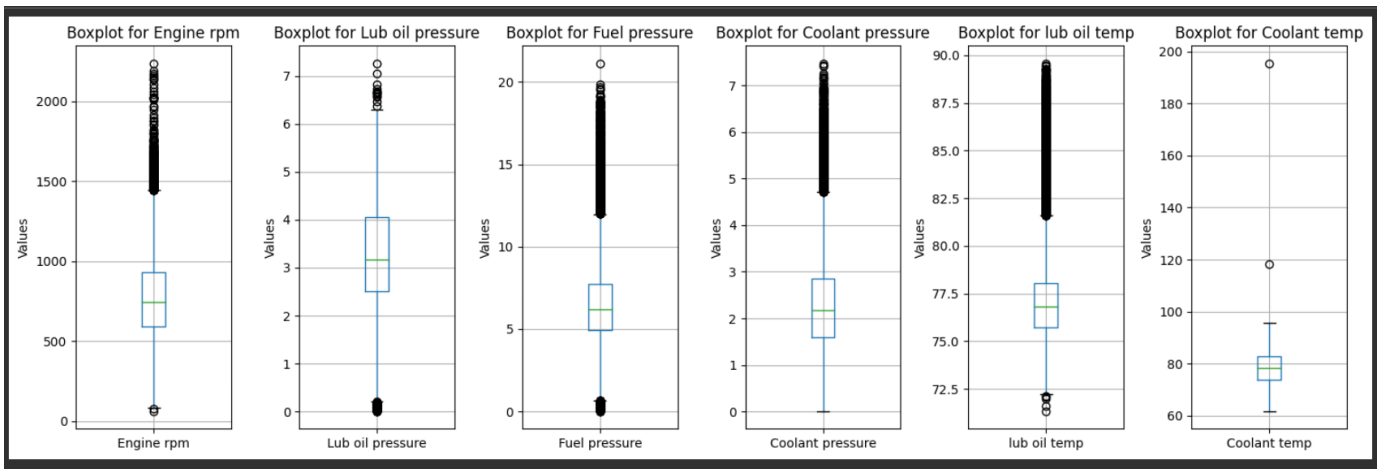
3. Στην συνέχεια δημιουργούμαι ένα καινούργιο **DataFrame** με την εντολή

   df =pd.DataFrame(features) η οποία δημιουργεί ένα νέο **DataFrame df** από το **features**.

4. Το επόμενο βήμα είναι να ορίσουμε δυο συναρτήσεις:

   detect_outliers_iqr για την ανίχνευση ακραίων τιμών με χρήση της ενδοτεταρτημοριακής απόστασης **(IQR)**.

   remove_outliers_iqr για την αφαίρεση ακραίων τιμών με χρήση της ίδιας μεθόδου Το **IQR** χρησιμοποιείται για τη μέτρηση της μεταβλητότητας διαιρώντας ένα σύνολο δεδομένων σε τεταρτημόρια.



Σχήμα 2.6: Outliers

Με αυτόν τον τρόπο για κάθε στήλη του **DataFrame df**, ανιχνεύονται οι ακραίες τιμές και, αν υπάρχουν, εκτυπώνονται και αφαιρούνται από το **DataFrame**.

5. Στην συνέχεια ορίζουμε δυο συναρτήσεις για να ανίχνευσουμαι και αφαίρεσουμαι ξανά τις ακραίες τιμές χρησιμοποιώντας **z-score**.

   Ορίζουμαι τις συναρτήσεις:

   detect_outliers_zscore για την ανίχνευση ακραίων τιμών με χρήση του **z-score**.

   remove_outliers_zscore για την αφαίρεση ακραίων τιμών με χρήση της ίδιας μεθόδου.

   Το **z-score** είναι ένας δείκτης που μετρά πόσες τυπικές αποκλίσεις μια τιμή απέχει από το μέσο όρο των δεδομένων. Αυτό προσφέρει μια καθολική μέτρηση για το πώς η τιμή ενός δεδομένου συγκρίνεται με το σύνολο του πληθυσμού.

$$z = \frac{X - \mu}{\sigma}$$

   - X = X είναι η τιμή του δεδομένου στοιχείου.
   - μ = μ είναι ο μέσος όρος του **dataset**.
   - σ = σ είναι η τυπική απόκλιση του **dataset**.

6. Και τέλος ανίχνευει και αφαίρει τις ακραίες τιμές με χρήση **z-score** για κάθε στήλη του **DataFrame df**, ανιχνεύοντας τις ακραίες τιμές και αν υπάρχουν, εκτυπώνονται και αφαιρούνται από το **DataFrame**.

2.4 Προεπεξεργασία δεδομένων

Σε αυτό το βήμα θα δούμε σύνθετες τεχνικές προεπεξεργασίας και επεξεργασίας για τη βελτίωση της απόδοσης των μοντέλων μηχανικής μάθησης που θα κατασκευαστούν αργότερα με αυτά τα δεδομένα. Αυτές οι μέθοδοι βοηθούν στην αντιμετώπιση συνήθων προκλήσεων όπως η ανισορροπία των κλάσεων, η επίδραση των εξωκείμενων τιμών και η σημασία της κλίμακας των δεδομένων.

2.4.1 Oversampling με τη χρήση της βιβλιοθήκης imblearn:

Εφαρμόζουμαι την μέθοδο **RandomOverSampler** για να ισορροπηθούν οι κλάσεις στο στόχο **target**. Αυτό βοηθά στην αποφυγή προκαταλήψεων προς την πιο συχνή κλάση σε ένα ανισορροπημένο **dataset**. Ο κώδικας παράγει δύο ισορροπημένες κλάσεις (0 και 1) με 12317 δείγματα η κάθε μία.

2.4.2 Διαχωρισμός σε σύνολο εκπαίδευσης και δοκιμής:

Χρησιμοποιούμαι την συνάρτηση **train_test_split** της **sklearn** για να διαχωρίσουμαι τα δείγματα σε σύνολο εκπαίδευσης (70%) και δοκιμής (30%), διατηρώντας την ισορροπία των κλάσεων χάρη στην παράμετρο **stratify**.

2.4.3 Κανονικοποίηση δεδομένων (Standard Scaling):

Χρησιμοποιούμαι **StandardScaler** για να κανονικοποιήσει τα δεδομένα, μετατρέποντας τη μέση τιμή σε 0 και την τυπική απόκλιση σε 1. Αυτό βοηθά στη βελτίωση της απόδοσης πολλών αλγορίθμων μηχανικής μάθησης.

2.4.4 Εκτύπωση στατιστικών μετά την κανονικοποίηση:

Οι μέσοι όροι και οι τυπικές αποκλίσεις για τα κανονικοποιημένα σύνολα δεδομένων είναι περίπου 0 και 1 αντίστοιχα για το σύνολο εκπαίδευσης, ενώ για το σύνολο δοκιμής, λόγω του μικρού δείγματος, υπάρχουν μικρές αποκλίσεις.

2.4.5 Εξαγωγή χαρακτηριστικών με Variance Threshold:

Χρησιμοποιούμαι το **VarianceThreshold** για την εξαγωγή χαρακτηριστικών. Αυτή η τεχνική αφαιρεί χαρακτηριστικά με διακύμανση κάτω από κάποιο καθορισμένο όριο (0.5 σε αυτή την περίπτωση), θεωρώντας ότι τα χαρακτηριστικά με πολύ χαμηλή διακύμανση προσφέρουν λιγότερη πληροφορία.

```
Διακυμάνσεις για κάθε χαρακτηριστικό πριν την εφαρμογή του VarianceThreshold:
Engine rpm          72347.752871
Lub oil pressure        1.022607
Fuel pressure           7.482560
Coolant pressure        1.112761
lub oil temp            9.901993
Coolant temp           38.157984
dtype: float64
Διακυμάνσεις για κάθε χαρακτηριστικό μετά την εφαρμογή του VarianceThreshold:
Engine rpm          1.000058
Lub oil pressure    1.000058
Fuel pressure       1.000058
Coolant pressure    1.000058
lub oil temp        1.000058
Coolant temp        1.000058
dtype: float64
```

Σχήμα 2.7: Variance threshold

3. ΒΗΜΑΤΑ ΠΡΟΕΠΕΞΕΡΓΑΣΙΑΣ

3.1 Β) Για την αναγνώριση σημάτων :

TRAIN MODEL/TEST MODEL

3.2 Μετατροπή σε Ασπρόμαυρη (Grayscale):

Η διαδικασία αρχίζει με τη μετατροπή των εικόνων από **RGB** σε ασπρόμαυρες **(grayscale)**. Αυτό γίνεται για να μειωθεί η πολυπλοκότητα των δεδομένων, καθώς η ασπρόμαυρη εικόνα απαιτεί λιγότερη υπολογιστική ισχύ και μνήμη για την επεξεργασία. Η μετατροπή γίνεται με τη χρήση της συνάρτησης **cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)**, η οποία απομακρύνει τις χρωματικές πληροφορίες και αφήνει μόνο τις φωτεινότητες των **pixels**. Πιο αναλυτικά η μετατροπή εικόνων από χρωματικές **(RGB)** σε ασπρόμαυρες **(grayscale)** αποτελεί ένα κρίσιμο βήμα προεπεξεργασίας στην επεξεργασία εικόνων και ιδιαίτερα στην εκπαίδευση συνελικτικών νευρωνικών δικτύων **(CNN)** για εφαρμογές όπως η αναγνώριση τροχαίων σημάτων.

3.3 Τεχνική Μετατροπής:

Η τεχνική χρησιμοποιεί τη συνάρτηση **cv2.cvtColor** με την παράμετρο **cv2.COLOR_BGR2GRAY** της βιβλιοθήκης **OpenCV** για να μετατρέψει τις εικόνες από τρία χρωματικά κανάλια (Κόκκινο, Πράσινο, Μπλε) σε ένα κανάλι φωτεινότητας. Η μετατροπή αυτή πραγματοποιείται αφαιρώντας τις χρωματικές πληροφορίες και διατηρώντας μόνο τη φωτεινότητα των **pixels**, κάτι που απλοποιεί σημαντικά την εικόνα.

3.4 Σκοπός Μετατροπής:

Ο κύριος σκοπός για τη μετατροπή σε ασπρόμαυρη εικόνα είναι η μείωση της πολυπλοκότητας των δεδομένων. Στα **συνελικτικά νευρωνικά δίκτυα**, η είσοδος λιγότερων δεδομένων μπορεί να οδηγήσει σε ταχύτερη επεξεργασία και λιγότερες υπολογιστικές απαιτήσεις, αλλά επίσης σε γρηγορότερη εκπαίδευση και σε καλύτερη γενίκευση από το μοντέλο.

3.5 Πλεονεκτήματα στην Επεξεργασία Εικόνων:

3.5.1 Μείωση Διαστάσεων:

Κάθε εικόνα απαιτεί λιγότερη μνήμη και υπολογιστική ισχύ για την αποθήκευση και επεξεργασία.

3.5.2 Εστίαση σε Δομικά Χαρακτηριστικά:

Αφαιρώντας τα χρώματα, το μοντέλο "μαθαίνει" να εστιάζει σε δομικά και μορφολογικά χαρακτηριστικά των εικόνων, όπως το σχήμα και την υφή, που είναι ουσιαστικά για την αναγνώριση τροχαίων σημάτων.

3.5.3 Βελτιωμένη Αντίθεση και Ορατότητα:

Σε συνδυασμό με την ισοστάθμιση ιστογράμματος, η ασπρόμαυρη εικόνα μπορεί να παρουσιάσει καλύτερη αντίθεση και ευκρίνεια στα διακριτικά στοιχεία.

Η επιλογή να μετατρέψουμε εικόνες σε ασπρόμαυρες για την εκπαίδευση **CNN** στην αναγνώριση τροχαίων σημάτων αντανακλά μια στρατηγική που ισορροπεί μεταξύ απόδοσης και ακρίβειας, εξασφαλίζοντας ότι το μοντέλο είναι όσο το δυνατόν πιο αποτελεσματικό και αποδοτικό στην επεξεργασία και την αναγνώριση των εικόνων.

3.6 Ισοστάθμιση Ιστογράμματος (Histogram Equalization):

Το επόμενο βήμα είναι η ισοστάθμιση του ιστογράμματος, η οποία βελτιώνει την αντίθεση της εικόνας. Αυτή η επεξεργασία καθιστά τα αντικείμενα και τις λεπτομέρειες στην εικόνα πιο διακριτικά και ευκολότερα αναγνωρίσιμα από το νευρωνικό δίκτυο. Η τεχνική αυτή είναι χρήσιμη ιδιαίτερα σε συνθήκες χαμηλού φωτισμού ή όταν οι εικόνες προέρχονται από διαφορετικές πηγές με διαφορετικά επίπεδα φωτεινότητας και αντίθεσης. Η ισοστάθμιση γίνεται με τη συνάρτηση **cv2.equalizeHist(img)**. Η ισοστάθμιση ιστογράμματος **(Histogram Equalization)** είναι μια σημαντική τεχνική στην επεξεργασία εικόνας που χρησιμοποιείται για να βελτιώσει την αντίθεση μιας εικόνας. Η εφαρμογή της στον κώδικα μέσω της συνάρτησης **cv2.equalizeHist(img)** της βιβλιοθήκης **OpenCV** συμβάλλει σημαντικά στην ανάδειξη των λεπτομερειών και των αντικειμένων που περιέχονται στις εικόνες. Ας εξετάσουμε αναλυτικά την διαδικασία της ισοστάθμισης ιστογράμματος.

3.7 Διαδικασία Ισοστάθμισης Ιστογράμματος:

Η ισοστάθμιση ιστογράμματος λειτουργεί με τον εξομαλυντικό της κατανομής των φωτεινοτήτων της εικόνας. Αυτό επιτυγχάνεται αναδιαρθρώνοντας την κατανομή των φωτεινοτήτων έτσι ώστε οι περιοχές με χαμηλότερη αντίθεση να αποκτήσουν μεγαλύτερη διακύμανση στις τιμές φωτεινότητας. Το αποτέλεσμα είναι μια εικόνα που έχει πιο ισορροπημένη κατανομή φωτεινότητας και ενισχυμένη αντίθεση.

3.8 Αιτιολόγηση Εφαρμογής της Ισοστάθμισης:

3.8.1 Βελτίωση Αντίθεσης:

Η κύρια λειτουργία της ισοστάθμισης ιστογράμματος είναι η βελτίωση της αντίθεσης. Αυτό είναι κρίσιμο για τις εφαρμογές αναγνώρισης εικόνων, όπου η ακριβής απεικόνιση των αντικειμένων μπορεί να επηρεαστεί από την ποιότητα της αρχικής εικόνας.

3.8.2 Ανάδειξη Λεπτομερειών:

Ενισχύοντας την αντίθεση, οι λεπτομέρειες που μπορεί να ήταν δυσδιάκριτες σε μια εικόνα με χαμηλή αντίθεση γίνονται πιο ξεκάθαρες και ευδιάκριτες.

3.8.3 Εξισορρόπηση Φωτεινότητας:

Η ισοστάθμιση βοηθά στην εξισορρόπηση των περιοχών υψηλής και χαμηλής φωτεινότητας, κάτι που είναι ωφέλιμο όταν οι εικόνες προέρχονται από διαφορετικά περιβάλλοντα φωτισμού.

3.8.4 Προσαρμογή σε Διαφορετικές Συνθήκες Φωτισμού:

Η τεχνική αυτή είναι ιδιαίτερα χρήσιμη σε εφαρμογές όπου οι εικόνες έχουν ληφθεί υπό διαφορετικές συνθήκες φωτισμού, βοηθώντας το μοντέλο να διατηρεί σταθερή απόδοση ανεξάρτητα από εξωτερικούς παράγοντες.

Με την ισοστάθμιση ιστογράμματος, η εικόνα που τροφοδοτείται στο νευρωνικό δίκτυο είναι πιο κατάλληλη για επεξεργασία, με ομοιόμορφη αντίθεση και φωτεινότητα, εξασφαλίζοντας έτσι ότι το δίκτυο μπορεί να "κατανοήσει" και να "μάθει" αποτελεσματικά τα χαρακτηριστικά από τις εικόνες που του προσφέρονται.

3.9 Κανονικοποίηση (Normalization)

Το τελευταίο βήμα της προεπεξεργασίας είναι η κανονικοποίηση των τιμών των εικόνων σε μια κλίμακα από 0 έως 1. Αυτό βοηθά στην ομοιόμορφη εκπαίδευση του δικτύου, καθώς εξασφαλίζει ότι οι τιμές των εισόδων είναι κανονικοποιημένες και δεν θα προκαλέσουν μη-βέλτιστες αλλαγές στις παραμέτρους του δικτύου λόγω υπερβολικά υψηλών ή χαμηλών τιμών. Η κανονικοποίηση γίνεται μετατρέποντας την εικόνα σε **float32** και διαιρώντας τις τιμές της με 255.Αυτά τα βήματα προεπεξεργασίας παρέχουν στο μοντέλο μια καθαρή, κανονικοποιημένη και αναγνωρίσιμη μορφή των εικόνων για να διευκολύνουν την αποτελεσματική εκπαίδευση και αναγνώριση των τροχαίων σημάτων. Η κανονικοποίηση των τιμών των εικόνων σε μια κλίμακα από 0 έως 1 είναι ένα ζωτικό βήμα στην προεπεξεργασία εικόνων πριν την εκπαίδευση των συνελικτικών νευρωνικών δικτύων **(CNN)**. Ας εξετάσουμε αναλυτικά το πώς γίνεται αυτή η διαδικασία, γιατί είναι σημαντική, και τι επιπτώσεις έχει στην επίδοση του μοντέλου.

3.10 Διαδικασία Κανονικοποίησης :

Η κανονικοποίηση στην επεξεργασία εικόνων περιλαμβάνει τη μετατροπή των τιμών **pixel** από το εύρος 0-255 (ο τυπικός εύρος τιμών για **8-bit** εικόνες) σε τιμές μεταξύ 0 και 1. Αυτό επιτυγχάνεται μετατρέποντας το **datatype** της εικόνας σε **float32** και διαιρώντας κάθε τιμή **pixel** με 255. Η συνάρτηση στον κώδικα είναι η εξής:

κανονικοποίηση τιμών σε 0-1 από 0-255

img = img.astype("float32")

img = img / 255

return img

3.11 Σημασία της Κανονικοποίησης

3.11.1 Βελτίωση Σύγκλισης Μάθησης:

Κανονικοποιώντας τις τιμές των εισόδων, βοηθάμε το μοντέλο να συγκλίνει πιο γρήγορα κατά τη διαδικασία εκπαίδευσης. Νευρωνικά δίκτυα με τυχαία αρχικοποιημένα βάρη αντιδρούν καλύτερα σε μικρότερες και πιο ομοιόμορφες τιμές εισόδου.

3.11.2 Αποφυγή Κορεσμού Νευρώνων:

Όταν οι τιμές εισόδου είναι πολύ υψηλές, υπάρχει κίνδυνος οι νευρώνες στα επίπεδα του δικτύου να κορεστούν. Αυτό σημαίνει ότι οι νευρώνες μπορεί να ενεργοποιούνται σχεδόν πάντα, χάνοντας τη δυνατότητα του δικτύου να διακρίνει αποτελεσματικά μοτίβα και χαρακτηριστικά στα δεδομένα.

3.11.3 Ομοιογενής Επίδραση Χαρακτηριστικών:

Κανονικοποιώντας τις τιμές των εικόνων, διασφαλίζουμε ότι κανένα συγκεκριμένο χαρακτηριστικό (όπως η φωτεινότητα ή το χρώμα) δεν θα έχει υπερβολικά μεγάλη επίδραση στην εκπαίδευση λόγω των υψηλότερων αριθμητικών τιμών του.

Αυτές οι ιδιότητες καθιστούν την κανονικοποίηση ένα ζωτικό κομμάτι της προεπεξεργασίας σε οποιαδήποτε διαδικασία μηχανικής μάθησης που περιλαμβάνει εικόνες, ιδιαίτερα όταν στόχος είναι η αναγνώριση και η κατηγοριοποίηση αντικειμένων.

4. ΠΛΗΡΟΦΟΡΙΕΣ ΣΥΝΟΛΩΝ ΔΕΔΟΜΕΝΩΝ

4.1 Α) Για τον κινητήρα :

Η ανάλυση των χαρακτηριστικών του συνόλου δεδομένων περιλάμβανε διάφορες τεχνικές οπτικοποίησης για την κατανόηση των σχέσεων και των κατανομών των μετρήσεων από τους αισθητήρες του κινητήρα. Η καταγραφή των στηλών και των περιεχομένων τους, καθώς και η επιλογή και ανάλυση των χαρακτηριστικών, βοήθησαν στην προετοιμασία και καθαρισμό των δεδομένων, και στην αντιμετώπιση της ανισορροπίας των κλάσεων πριν από την εκπαίδευση των μοντέλων.

Η χρήση εργαλείων όπως το **heatmap**, τα **pairplots**, τα **histograms**, τα **boxplots** και τα **scatter plot matrices** επέτρεψε μια λεπτομερή ανάλυση και οπτικοποίηση των δεδομένων. Το **heatmap** παρείχε μια διαισθητική απεικόνιση των συσχετίσεων μεταξύ των χαρακτηριστικών, ενώ τα **histograms** βοήθησαν στην κατανόηση της κατανομής των τιμών των δεδομένων. Τα **boxplots** και τα **violin plots** αποκάλυψαν την κεντρική τάση, τη διασπορά και τις ακραίες τιμές των χαρακτηριστικών, ενώ το **scatter plot matrix** επέτρεψε την οπτικοποίηση των αλληλεπιδράσεων μεταξύ πολλών χαρακτηριστικών.

Η αντιμετώπιση της ανισορροπίας των κλάσεων ήταν κρίσιμη για την βελτίωση της απόδοσης των μοντέλων μηχανικής μάθησης. Μέσα από αυτήν την αναλυτική διαδικασία, κατέστη δυνατή η λεπτομερής κατανόηση των δεδομένων και η κατάλληλη προετοιμασία τους για την επόμενη φάση της ανάλυσης.

4.2 Ανάλυση Χαρακτηριστικών:

4.2.1 Καταγραφή Στηλών και Περιεχομένων

Το σύνολο δεδομένων που αναλύθηκε περιλαμβάνει μετρήσεις από διάφορους αισθητήρες ενός κινητήρα. Οι στήλες του συνόλου δεδομένων είναι οι εξής:

- **Engine rpm:** Οι στροφές του κινητήρα ανά λεπτό.
- **Lub oil pressure:** Η πίεση του λαδιού λίπανσης.
- **Fuel pressure:** Η πίεση του καυσίμου
- **Coolant pressure:** Η πίεση του ψυκτικού υγρού.
- **Lub oil temp:** Η θερμοκρασία του λαδιού λίπανσης.
- **Coolant temp:** Η θερμοκρασία του ψυκτικού υγρού.

Η μεταβλητή στόχος είναι η **Engine Condition**, η οποία αναπαριστά την κατάσταση του κινητήρα και είναι δυαδική **(0 ή 1)**.

4.2.2 Επιλογή Στηλών για Ανάλυση

Για την ανάλυση επιλέχθηκαν όλες οι διαθέσιμες στήλες των χαρακτηριστικών. Αυτές οι στήλες περιέχουν τις μετρήσεις από τους διάφορους αισθητήρες και είναι κρίσιμες για την κατανόηση της λειτουργίας και της κατάστασης του κινητήρα.

4.3 Διαδικασία Επιλογής

Η διαδικασία επιλογής των στηλών και η ανάλυσή τους περιλαμβάνει τα εξής βήματα:

4.3.1 Υπολογισμός και Οπτικοποίηση Συσχετίσεων:

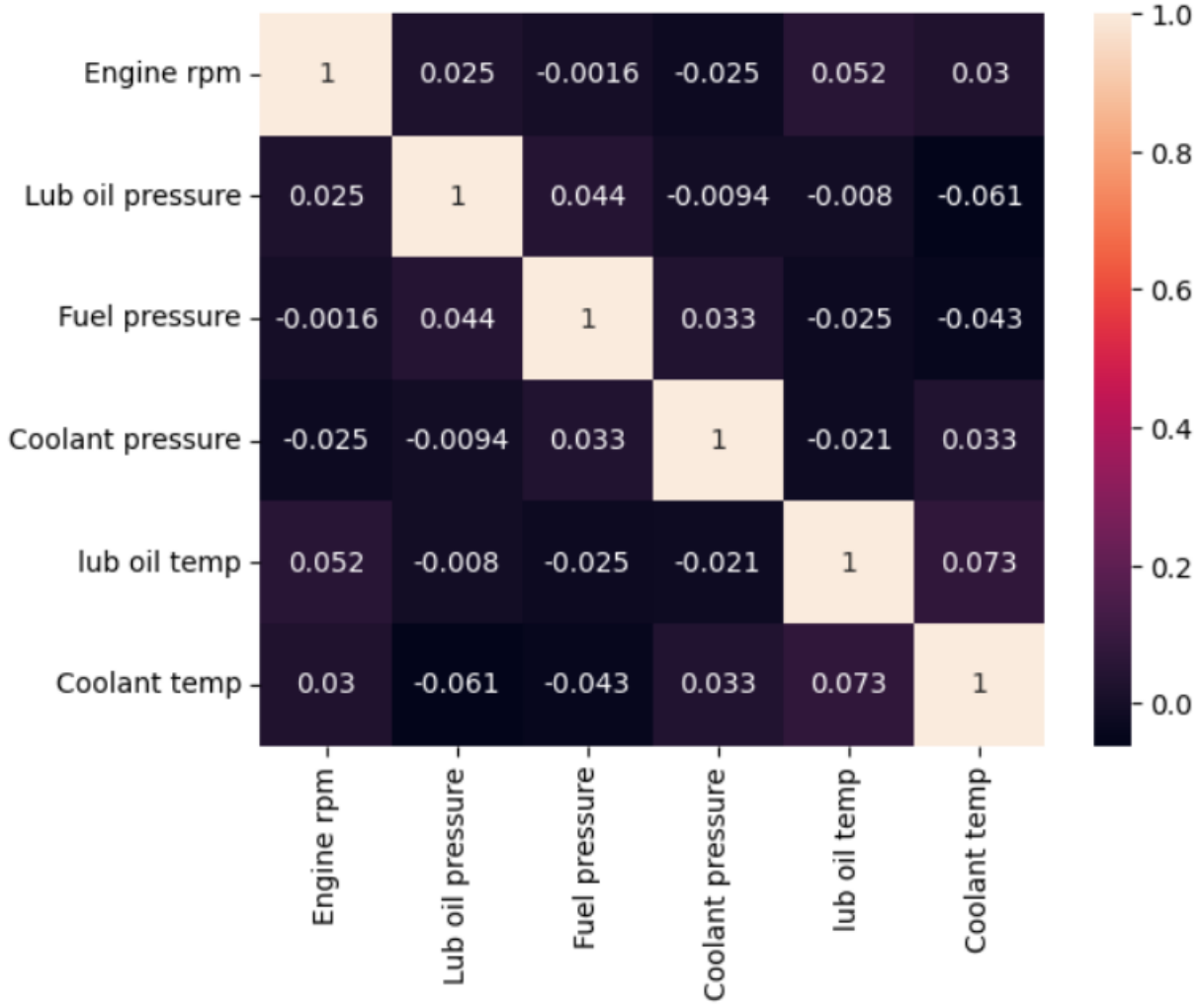
Ο υπολογισμός συσχετίσεων χρησιμοποιεί την συνάρτηση **.corr()** της **Pandas** για να υπολογιστεί η συσχέτιση μεταξύ των συνεχών χαρακτηριστικών του **DataFrame features**.

Η οπτικοποίηση με **heatmap:**

Χρησιμοποιείται η συνάρτησηheatmap της **Seaborn** για να απεικονιστούν οι συσχετίσεις μεταξύ των χαρακτηριστικών.

Τα χρώματα δείχνουν το μέγεθος της συσχέτισης, και οι τιμές είναι επίσης αναγραφόμενες. Το **heatmap** παρέχει μια γρήγορη και διαισθητική επισκόπηση του πόσο στενά συνδέονται τα χαρακτηριστικά μεταξύ τους.

Οι έντονες χρωματικές διαφοροποιήσεις βοηθούν στον εντοπισμό ισχυρών συσχετίσεων ή ανεξαρτησιών μεταξύ των μεταβλητών.



Σχήμα 4.1: Heatmap

Η εικόνα παραπάνω απεικονίζει έναν πίνακα συσχέτισης για διάφορες μεταβλητές κινητήρα, με χρήση **heatmap**. Η συσχέτιση κυμαίνεται από **-1 έως 1**, όπου οι θετικές τιμές υποδηλώνουν θετική συσχέτιση και οι αρνητικές αρνητική συσχέτιση. Οι περισσότερες μεταβλητές δείχνουν πολύ χαμηλή έως μηδενική συσχέτιση μεταξύ τους, ενώ ορισμένες έχουν ήπια θετική συσχέτιση (π.χ., η σχέση μεταξύ **"Lub oil temp" και "Coolant temp"**). Συνολικά, οι συσχετίσεις μεταξύ των μεταβλητών είναι γενικά πολύ χαμηλές, εκτός από μερικές εξαιρέσεις με ελαφρώς θετική συσχέτιση. Αυτό υποδηλώνει ότι οι μεταβλητές αυτές λειτουργούν αρκετά ανεξάρτητα μεταξύ τους για τα συγκεκριμένα δεδομένα.

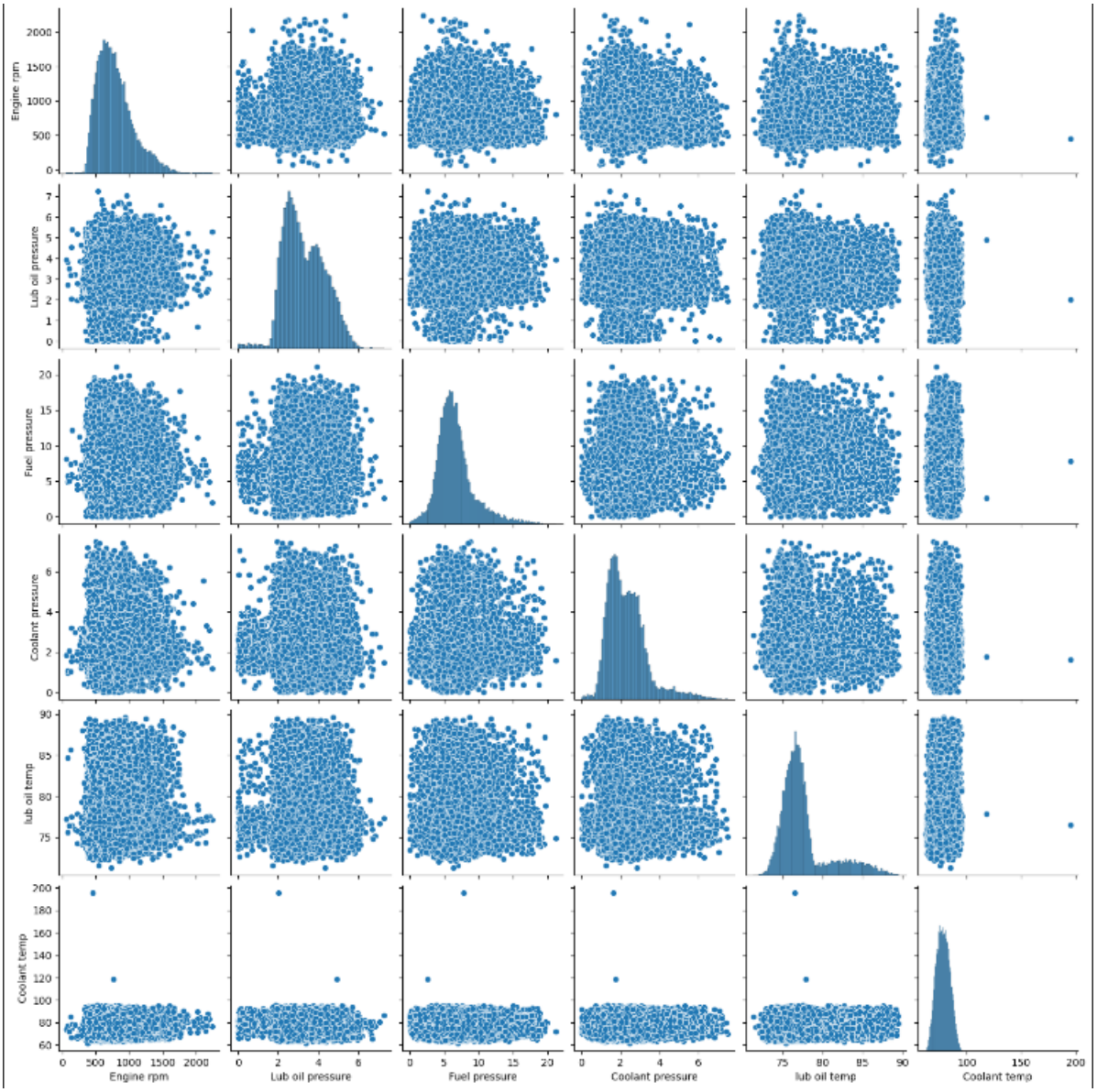
4.3.2 Οπτικοποίηση Διασποράς Δεδομένων:

Η χρήση **pairplot** δημιουργεί ένα **grid** γραφημάτων που δείχνει τη διασπορά κάθε ζεύγους χαρακτηριστικών και την κατανομή κάθε χαρακτηριστικού σε ιστόγραμμα. Το **pairplot** βοηθά στην κατανόηση των αλληλεπιδράσεων μεταξύ των χαρακτηριστικών και στην αναγνώριση πιθανών μοτίβων ή συσχετίσεων. Παρέχει διασπορές μεταξύ των χαρακτηριστικών καθώς και κατανομές τους σε διαγώνιες γραμμές, επιτρέποντας την ανίχνευση τόσο γραμμικών όσο και μη γραμμικών σχέσεων.

Η εικόνα παραπάνω χρησιμοποιείται για να αποκτήσετε μια καλή εποπτεία των δεδομένων και να καθορίσετε ποιες μεταβλητές μπορεί να αξίζει να εξεταστούν περαιτέρω για πιθανές σχέσεις. Αυτός ο τύπος γραφήματος επιτρέπει την απεικόνιση τόσο των κατανομών μεμονωμένων μεταβλητών όσο και των συσχετίσεων μεταξύ διαφόρων μεταβλητών. Στη διαγώνιο φαίνονται οι κατανομές των ατομικών μεταβλητών παρουσιαζόμενες μέσω ιστογραμμάτων, ενώ τα υπόλοιπα κελιά του πίνακα δείχνουν τα **scatterplots** των ζευγών μεταβλητών. Κάθε **scatterplot** δείχνει τη σχέση μεταξύ δύο μεταβλητών. Οι έντονες συγκεντρώσεις δεδομένων ή σαφείς τάσεις σε αυτά τα **plots** μπορούν να υποδηλώνουν ισχυρή συσχέτιση, ενώ πιο διάσπαρτα δεδομένα συνήθως υποδηλώνουν ασθενή ή μη συσχέτιση. Σε αυτή την εικόνα τα ιστογράμματα δείχνουν την κατανομή κάθε μεταβλητής. Για παράδειγμα, το ιστόγραμμα των **"Engine rpm"** δείχνει πώς τα δεδομένα συγκεντρώνονται κυρίως σε χαμηλότερες τιμές.Οι συσχετίσεις φαίνεται να είναι αρκετά ασθενείς για τις περισσότερες μεταβλητές, καθώς τα **scatterplots** δείχνουν δεδομένα διάσπαρτα πολύ ευρέως, χωρίς κάποιο ξεκάθαρο μοτίβο ή τάση.

4.3.3 Ιστογράμματα Χαρακτηριστικών:

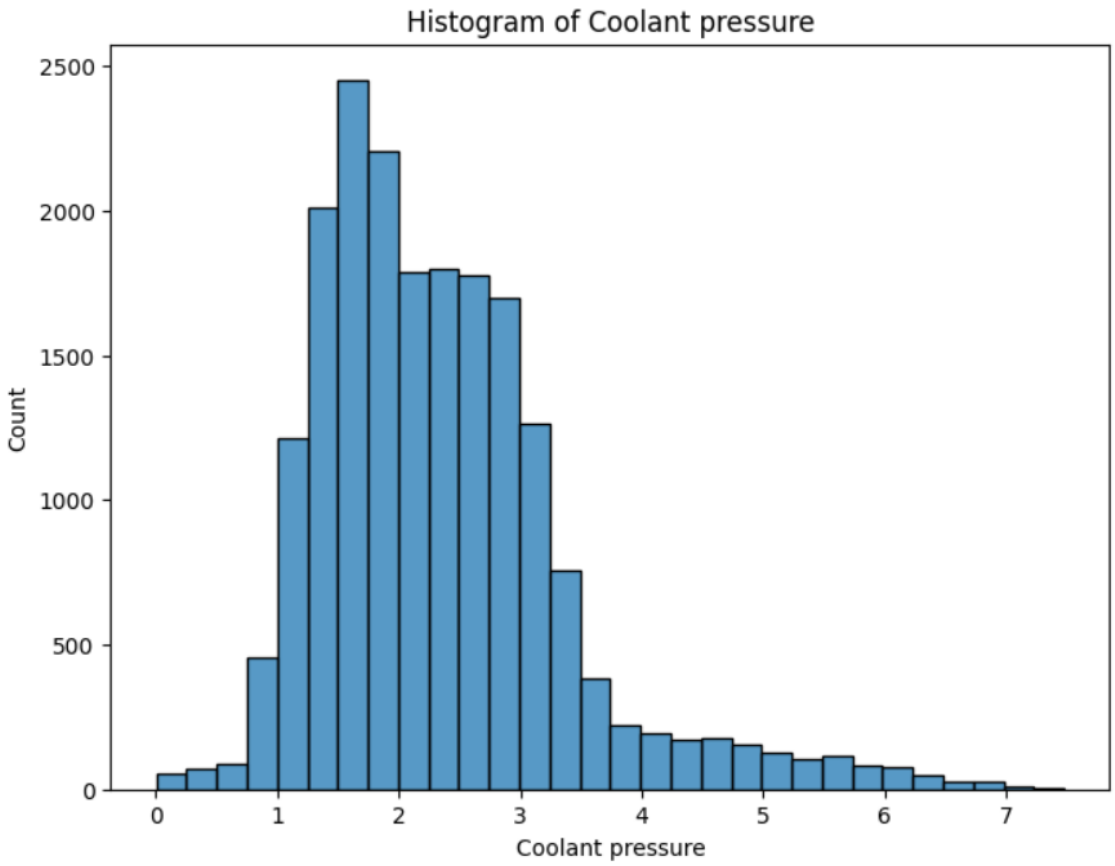
Η δημιουργία ιστογραμμάτων για κάθε χαρακτηριστικό, δημιουργείται ένα ιστόγραμμα που δείχνει την κατανομή των τιμών. Τα ιστογράμματα παρέχουν μια γραφική απεικόνιση της κατανομής των δεδομένων για κάθε χαρακτηριστικό, επιτρέποντας την ανίχνευση της πυκνότητας των δεδομένων σε διάφορες τιμές. Είναι χρήσιμα για την αναγνώριση των συχνοτήτων, των ασυμμετριών και της ύπαρξης πολλών κορυφών **(multi-modal distributions)**.



Σχήμα 4.2: Pairplot

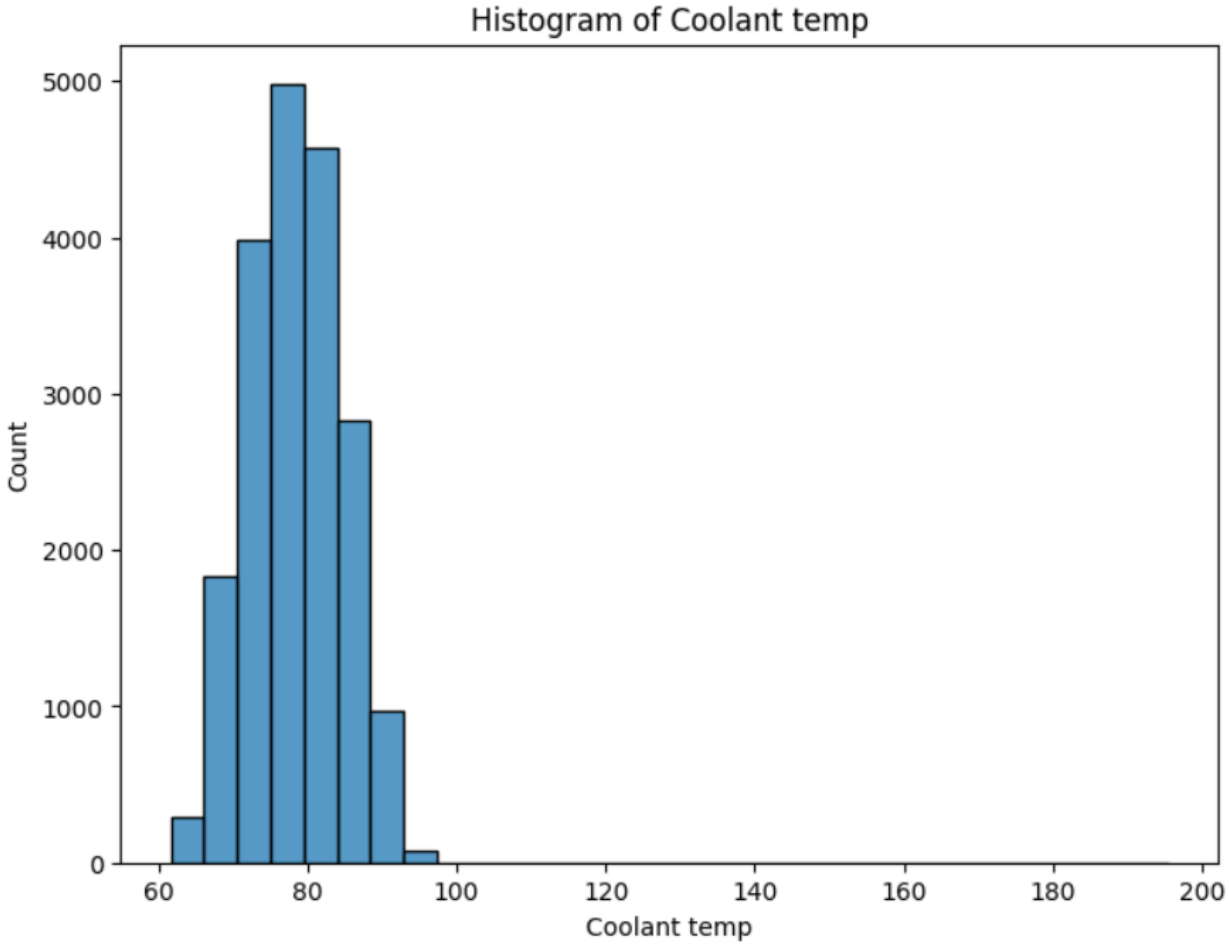
Το παραπανω ιστόγραμμα απεικονίζει την κατανομή των τιμών για την πίεση του ψυκτικού **(coolant pressure)** σε ένα σύνολο δεδομένων. Από το ιστόγραμμα φαίνεται ότι οι περισσότερες τιμές πίεσης του ψυκτικού συγκεντρώνονται κυρίως μεταξύ **2 και 4** μονάδων, με την κορύφωση της συχνότητας να εμφανίζεται γύρω στο **3**. Οι τιμές κλίνουν να μειώνονται καθώς απομακρυνόμαστε από αυτό το εύρος προς χαμηλότερες ή υψηλότερες τιμές. Αυτή η κατανομή δείχνει ότι μια σημαντική μερίδα του συνόλου δεδομένων διατηρεί μια σχετικά σταθερή πίεση ψυκτικού, ενώ μικρότερα ποσοστά των δεδομένων εμφανίζουν ακραίες τιμές, πολύ υψηλές ή πολύ χαμηλές, κάτι που μπορεί να υποδεικνύει εξαιρέσεις ή ανωμαλίες στα δεδομένα.

Το παραπάνω ιστόγραμμα δείχνει την κατανομή της θερμοκρασίας του ψυκτικού **(coolant temperature)** σε ένα σύνολο δεδομένων. Η κατανομή είναι συγκεντρωμένη κυρίως μεταξύ **60 και 100** βαθμών, με την κορύφωση της συχνότητας γύρω στους **80** βαθμούς. Η κατανομή φαίνεται να είναι κανονική, με την πλειοψηφία των τιμών να συγκεντρώνεται σε αυτό το εύρος και λιγότερες τιμές στα ακραία άκρα του εύρους.Η ανάλυση αυτού του ιστογράμματος δείχνει ότι το ψυκτικό του κινητήρα λειτουργεί σε θερμοκρασίες που είναι τυπικές για πολλές μηχανικές εφαρμογές, παρέχοντας μια ένδειξη για την κανονική λειτουργία



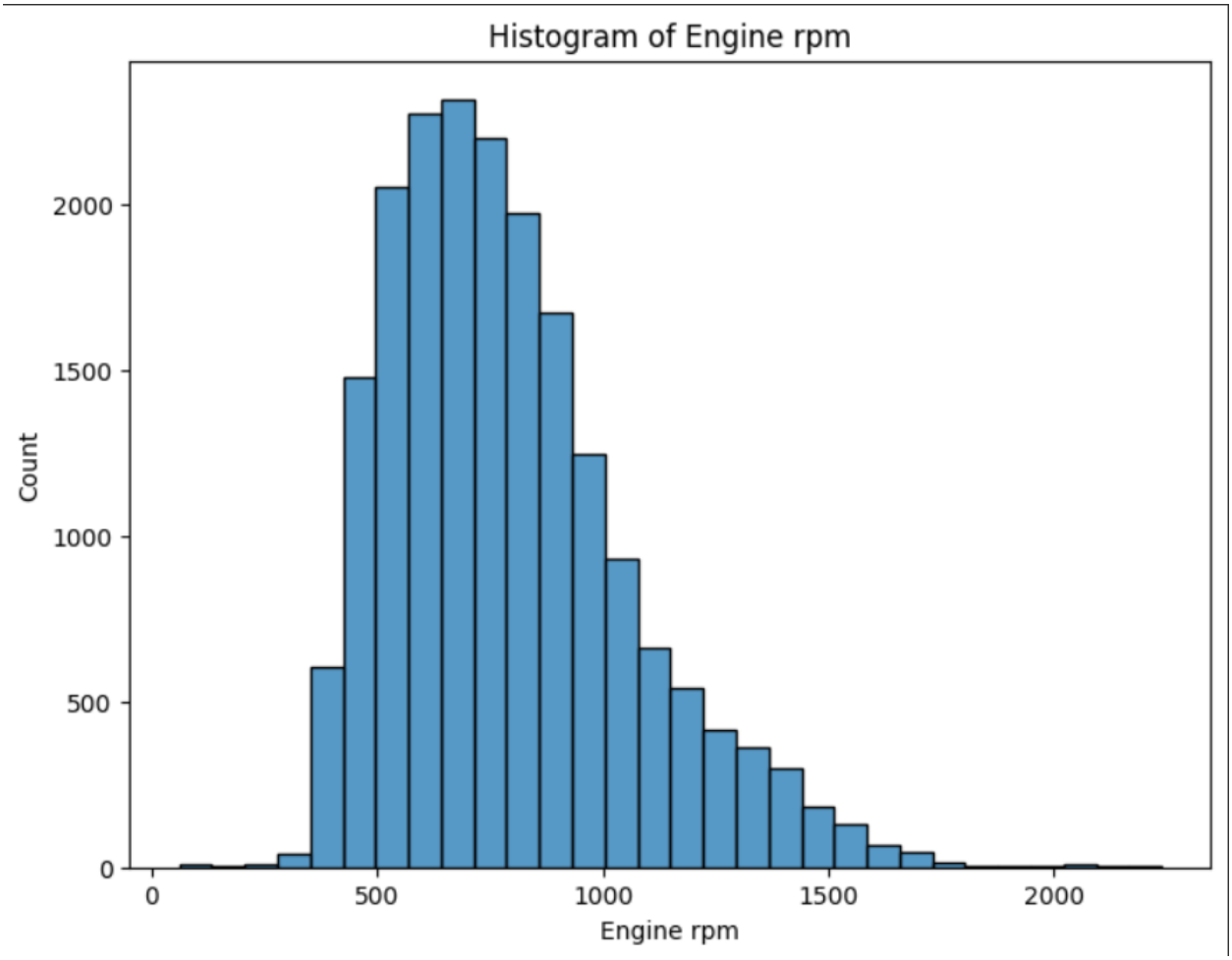
Σχήμα 4.3: Histogram coolant pressure

του κινητήρα. Ακραίες τιμές, ειδικά πολύ υψηλές, θα μπορούσαν να υποδεικνύουν πιθανά προβλήματα, όπως υπερθέρμανση.



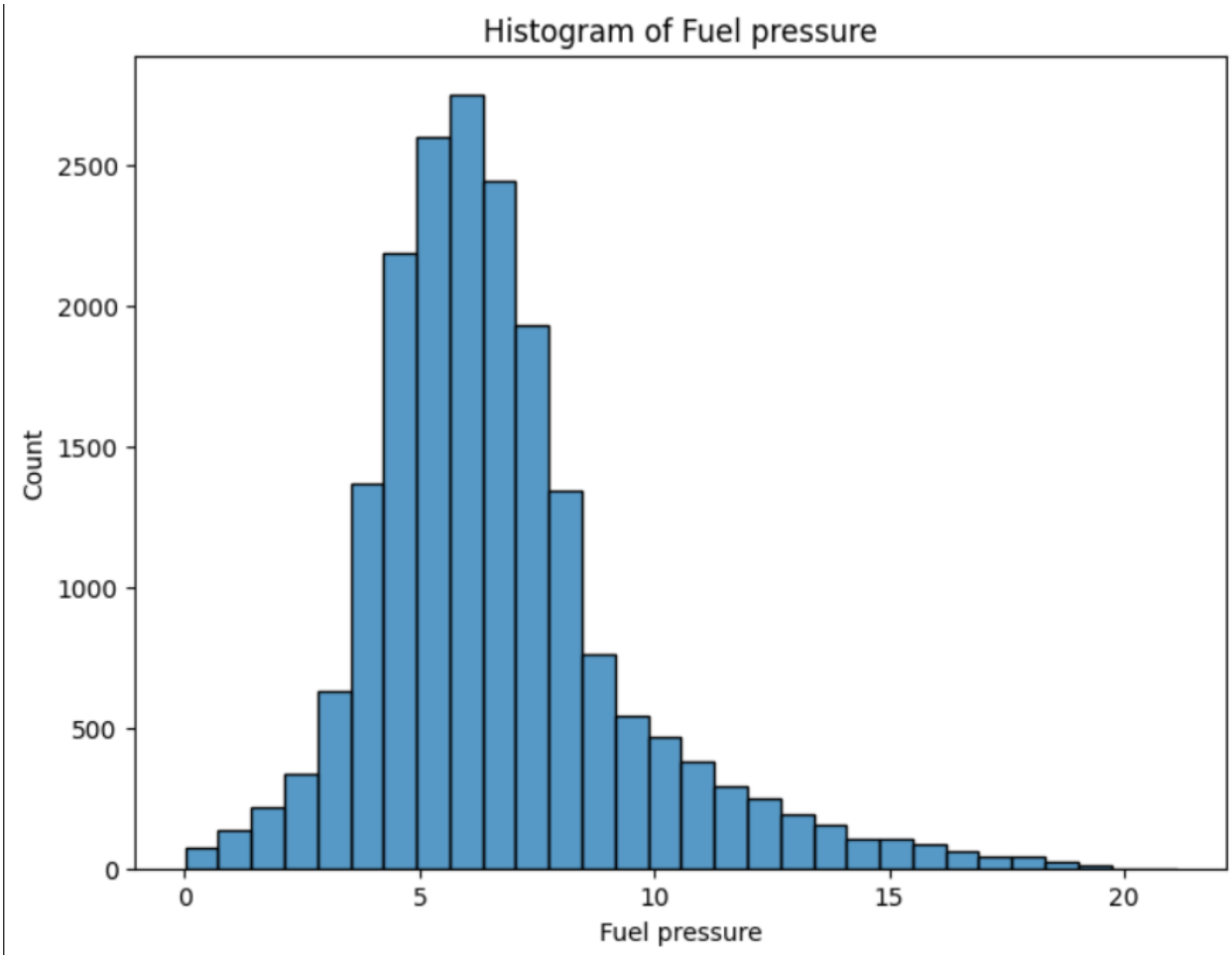
Σχήμα 4.4: Histogram coolant temp

Το παρακάτω ιστόγραμμα δείχνει την κατανομή των στροφών ανά λεπτό **(RPM)** ενός κινητήρα. Η κατανομή είναι καμπανοειδής και συγκεντρωμένη κυρίως γύρω από τις **1000** στροφές ανά λεπτό. Τα δεδομένα έχουν μια έντονη συγκέντρωση μεταξύ περίπου **500 και 800 RPM**, με μέγιστη συχνότητα κοντά στις **1000 RPM**. Το ιστόγραμμα δείχνει επίσης ότι υπάρχουν λιγότερες παρατηρήσεις καθώς αυξάνονται ή μειώνονται οι στροφές από αυτό το κεντρικό εύρος, με ταχύτητες κάτω από **500 και πάνω από 1500 RPM** να είναι σχετικά σπάνιες. Αυτή η κατανομή υποδηλώνει ότι οι περισσότεροι κινητήρες στο δείγμα λειτουργούν σε ένα σχετικά στενό εύρος στροφών, που είναι τυπικό για πολλά οχήματα κατά την κανονική λειτουργία.



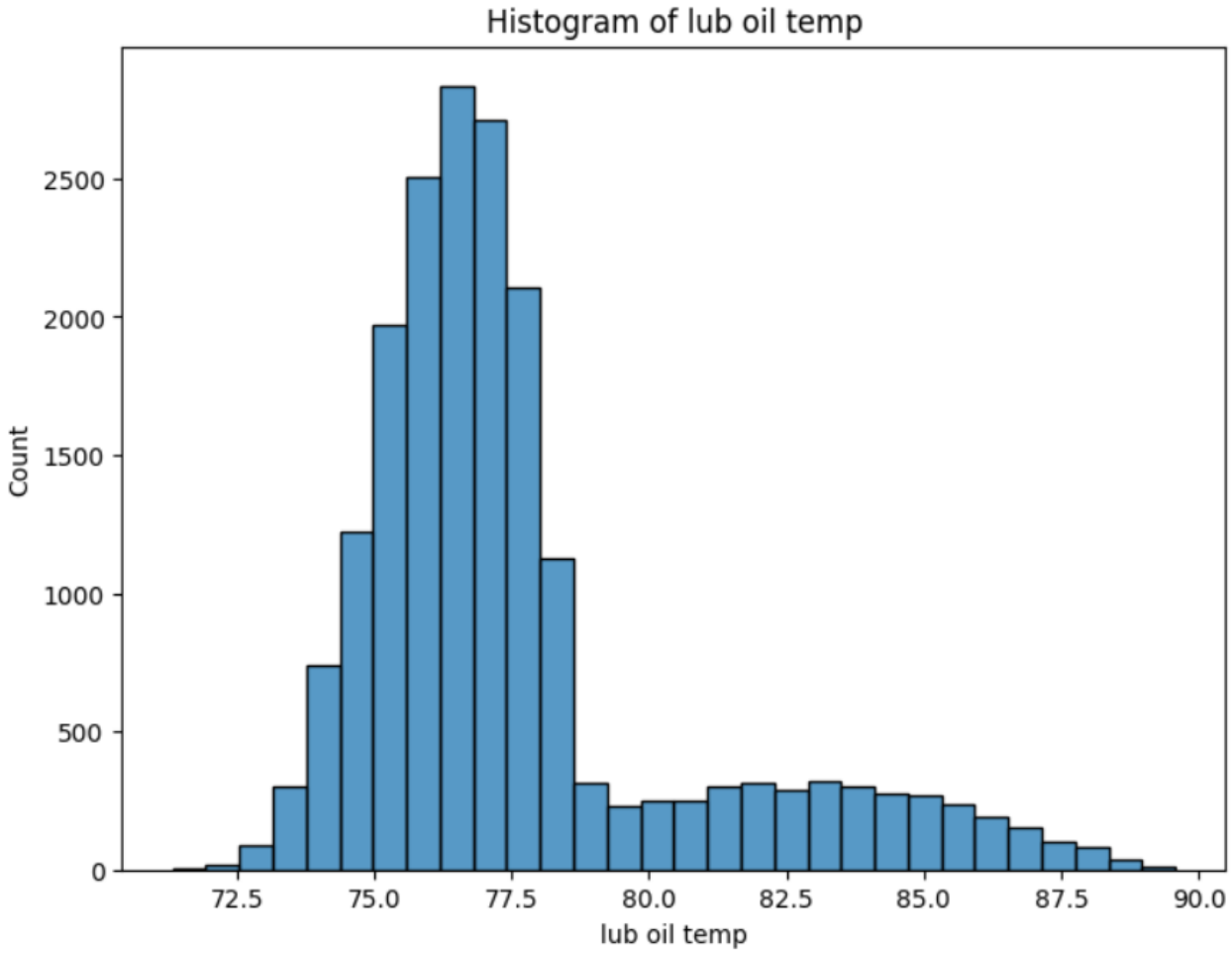
Σχήμα 4.5: Histogram engine rpm

Το παρακάτω ιστόγραμμα παρουσιάζει την κατανομή της πίεσης καυσίμου σε ένα σύνολο δεδομένων για κινητήρες. Η κατανομή της πίεσης καυσίμου είναι καμπανοειδής και κυρίως συγκεντρωμένη γύρω από τις **10** μονάδες πίεσης, υποδεικνύοντας ότι η πιο συχνή πίεση καυσίμου στα δείγματα είναι περίπου **10**. Η συχνότητα των τιμών μειώνεται όσο απομακρυνόμαστε από αυτό το κεντρικό σημείο προς χαμηλότερες ή υψηλότερες τιμές πίεσης. Ειδικότερα, οι τιμές πίεσης που κυμαίνονται από περίπου **7 έως 13** μονάδες είναι πολύ συχνές, ενώ τιμές κάτω από **5 ή πάνω από 15** είναι αρκετά σπάνιες. Αυτό δείχνει ότι η πίεση καυσίμου στους κινητήρες του δείγματος τείνει να είναι σχετικά σταθερή με κάποια μικρή μεταβλητότητα γύρω από μια τυπική τιμή, υποδηλώνοντας ότι η πίεση καυσίμου ελέγχεται στενά στις περισσότερες λειτουργικές καταστάσεις του κινητήρα.



Σχήμα 4.6: Histogram fuel pressure

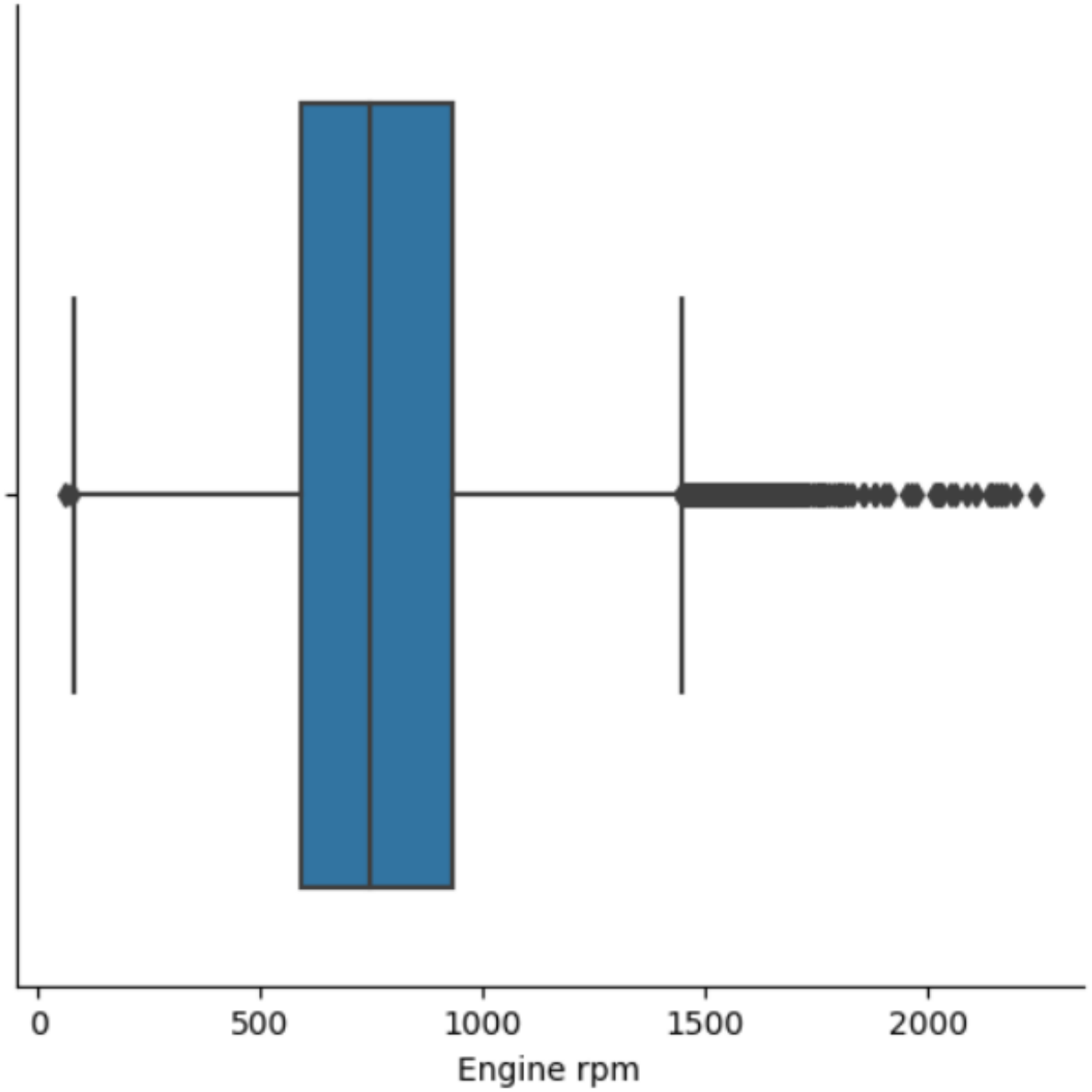
Το παρακάτω ιστόγραμμα απεικονίζει την κατανομή των τιμών θερμοκρασίας λιπαντικού λαδιού σε κινητήρες. Η κατανομή είναι κεντρικά συγκεντρωμένη γύρω από τις **77.5** βαθμούς Κελσίου, δείχνοντας ότι η πιο συνηθισμένη θερμοκρασία λιπαντικού λαδιού είναι περίπου σε αυτό το εύρος.Η κατανομή έχει σχήμα καμπάνας, με τη συγκέντρωση των δεδομένων στο κεντρικό εύρος, και εμφανίζει συμμετρία γύρω από την κορυφή της καμπάνας. Υπάρχει ένας σχετικά υψηλός αριθμός μετρήσεων μεταξύ **75 έως 80** βαθμών Κελσίου, με τις τιμές να μειώνονται καθώς απομακρύνονται από αυτό το κεντρικό εύρος. Οι τιμές κάτω από **72.5** και πάνω από **87.5** βαθμούς Κελσίου είναι σπάνιες, δείχνοντας ότι οι θερμοκρασίες λιπαντικού λαδιού διατηρούνται κυρίως μέσα σε αυτά τα όρια κατά τη λειτουργία του κινητήρα.



Σχήμα 4.7: Histogram lub oil temp

4.3.4 Διαγράμματα Κουτιών και Βιολιών:

Η δημιουργία **Boxplot** για το **Engine rpm** δημιουργεί ένα **boxplot** για να απεικονιστεί η κατανομή των τιμών της στήλης **Engine rpm**. Τα **boxplots** είναι χρήσιμα για την αναγνώριση ακραίων τιμών **(outliers)** και την κατανόηση της κατανομής των δεδομένων. Δείχνουν την κεντρική τάση και τη διασπορά των δεδομένων, καθώς και τυχόν ασυμμετρίες.



Σχήμα 4.8: Box plot engine rpm.png

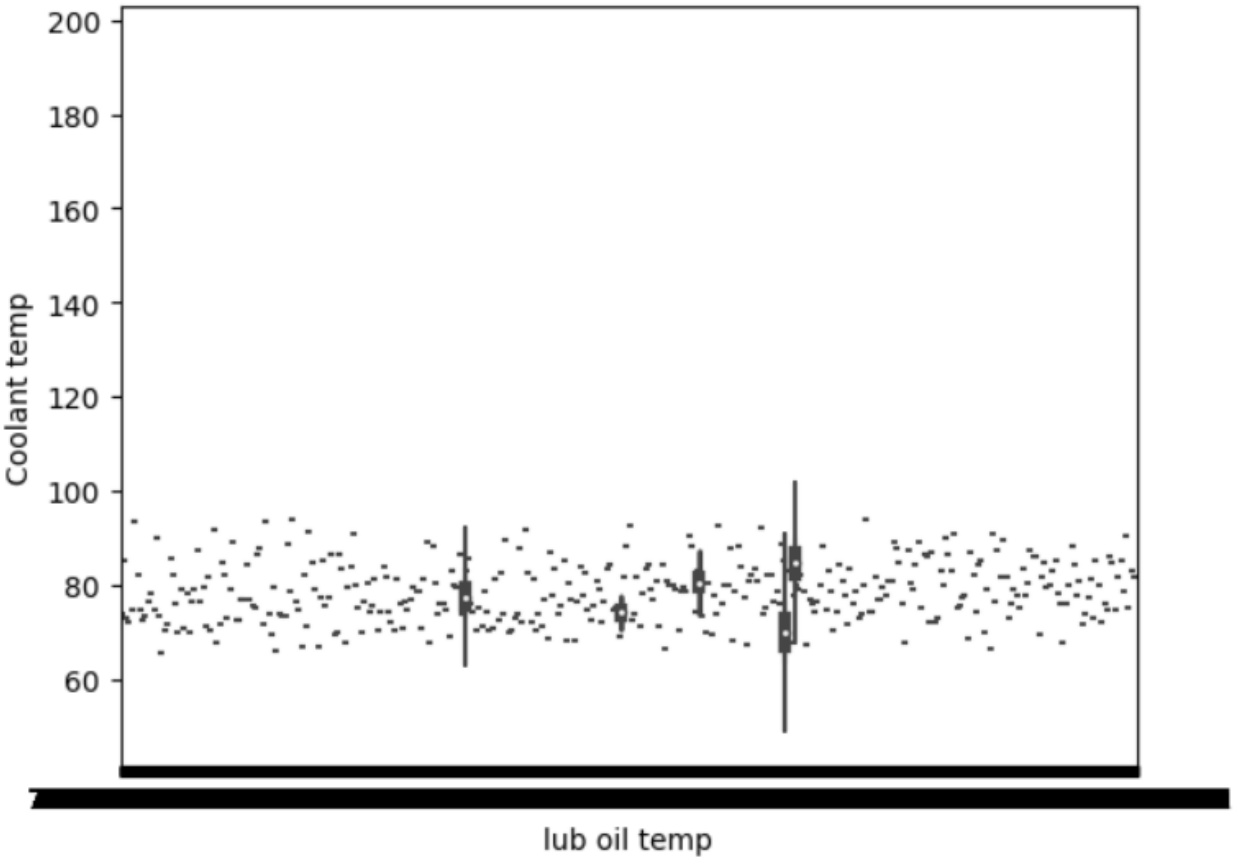
Η παραπάνω εικόνα που βλέπουμε είναι ένα **boxplot** (διάγραμμα κουτιού), που παρουσιάζει την κατανομή των ταχύτητων στροφών κινητήρα **(engine rpm)** σε ένα σύνολο δεδομένων. Το **boxplot** χρησιμοποιείται για να δείξει την κεντρική τάση και τη διασπορά των δεδομένων καθώς και για να εντοπίσει πιθανές ακραίες τιμές **(outliers)**. Ας αναλύσουμε τα στοιχεία του διαγράμματος:

1. **Κεντρικό Κουτί:**Το κεντρικό κουτί του διαγράμματος εμφανίζει τα τρία τεταρτημόρια της κατανομής δεδομένων:
    - **Κάτω άκρο του κουτιού:** Το πρώτο τεταρτημόριο **(Q1)**, που είναι η τιμή κάτω από την οποία βρίσκεται το **25%** των τιμών.
    - **Μεσαία γραμμή του κουτιού:** Η διάμεσος **(median)**, δηλαδή η τιμή που διαιρεί τη διανομή στο μέσο.
    - **Άνω άκρο του κουτιού:** Το τρίτο τεταρτημόριο **(Q3)**, πάνω από το οποίο βρίσκεται το **25%** των τιμών.

2. **Γραμμες (Whiskers):** Οι Γραμμές εκτείνονται από το κάτω και το άνω άκρο του κουτιού προς την ελάχιστη και μέγιστη τιμή αντίστοιχα, εξαιρουμένων των ακραίων τιμών. Εδώ βλέπουμε ότι οι μέγιστες τιμές είναι πολύ κοντά στο **Q3**, δηλαδή η μεγαλύτερη πυκνότητα των δεδομένων βρίσκεται στη μεσαία περιοχή των ταχύτητων στροφών.

3. **Ακραίες Τιμές (Outliers):** Οι μικροί μαύροι κύκλοι δείχνουν ακραίες τιμές που δεν εμπίπτουν στα κανονικά όρια των **whiskers**. Αυτές οι τιμές είναι σημαντικά υψηλότερες ή χαμηλότερες από το κύριο σώμα των δεδομένων και μπορεί να θεωρηθούν ως ανωμαλίες. Το διάγραμμα αυτό είναι πολύ χρήσιμο για την αξιολόγηση της διακύμανσης καθώς και της συχνότητας των τιμών στις ταχύτητες στροφών του κινητήρα, και μπορεί να βοηθήσει στην κατανόηση πώς αυτές οι τιμές διανέμονται σε σχέση με την κατάσταση του κινητήρα.

Δημιουργία **Violin Plot** για το **Lub oil temp** και **Box Plot** για **Coolant temp:** Το **violin plot** χρησιμοποιείται για να απεικονιστούν οι κατανομές των τιμών των δύο αυτών χαρακτηριστικών και οι σχέσεις τους. Συνδυάζει τα χαρακτηριστικά των **boxplots** και των **density plots**, προσφέροντας μια λεπτομερή εικόνα της κατανομής των δεδομένων. Τα **violin plots** είναι χρήσιμα γιατί δείχνουν την πυκνότητα των δεδομένων σε διάφορα επίπεδα, επιτρέποντας την αναγνώριση περιοχών με υψηλή ή χαμηλή συγκέντρωση τιμών.

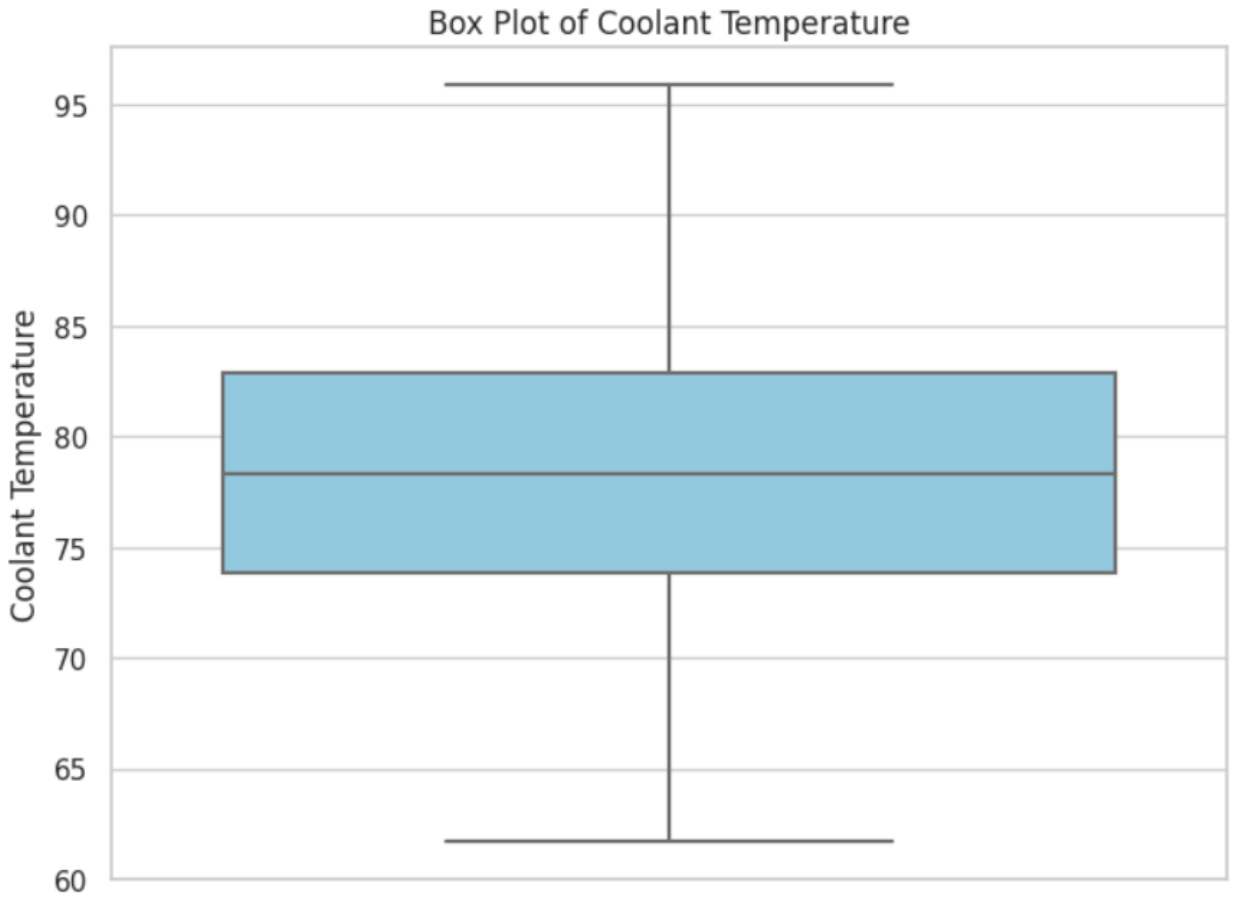


Σχήμα 4.9: Violin plot

Το διάγραμμα παραπάνω είναι ένα **Violin plot** που δείχνει τη σχέση μεταξύ της θερμοκρασίας του λιπαντικού του κινητήρα **(lub oil temp)** και της θερμοκρασίας ψύξης **(coolant temp)**. Ο άξονας **x** δείχνει τις τιμές της θερμοκρασίας του λιπαντικού σε βαθμούς Κελσίου, ενώ ο άξονας **y** δείχνει τις τιμές της θερμοκρασίας ψύξης επίσης σε βαθμούς Κελσίου. Τα δεδομένα είναι διασκορπισμένα στο διάγραμμα, επικεντρωμένα κυρίως σε μία κεντρική τιμή για τη θερμοκρασία λιπαντικού περί τους **75 με 85** βαθμούς Κελσίου. Οι αντίστοιχες τιμές θερμοκρασίας ψύξης φαίνεται να ποικίλλουν από περίπου **60 έως 120** βαθμούς Κελσίου, με την πλειοψηφία των τιμών να είναι συγκεντρωμένη περί τους **80 με 100** βαθμούς Κελσίου. Στο κέντρο του διαγράμματος υπάρχει μια μαύρη κάθετη γραμμή **(box plot)** που απεικονίζει τη διακύμανση και την κεντρική τάση των τιμών θερμοκρασίας ψύξης για τις τυπικές τιμές θερμοκρασίας λιπαντικού. Αυτή η γραμμή δείχνει το **median**, το εύρος και τις ακραίες τιμές **(outliers)**, επιτρέποντας μια γρήγορη οπτική εκτίμηση της κατανομής και της σχετικής συσχέτισης μεταξύ αυτών των δύο μεταβλητών.Οι πιο σκούρες κουκίδες αντιπροσωπεύουν πιθανώς ακραίες τιμές ή ανωμαλίες που μπορεί να χρειαστούν περαιτέρω ανάλυση για να κατανοήσουμε τους λόγους της απόκλισής τους από τα υπόλοιπα δεδομένα.

Το box plot παρακάτω παρέχει μια σαφή οπτική επισκόπηση της κατανομής της θερμοκρασίας του ψυκτικού **(coolant temperature)** σε αυτό το σύνολο δεδομένων, δείχνοντας που βρίσκεται η πλειοψηφία των τιμών και πόσο συμπαγής είναι η κατανομή. **Κύριο πλαίσιο (Box):** Το κεντρικό πλαίσιο καλύπτει το διάστημα από το **25ο εως το 75ο** εκατοστημόριο **(Q1 έως Q3)**, γνωστό και ως διάμεσο εύρος. Η θερμοκρασία εντός αυτού του διαστήματος κυμαίνεται περίπου από **75 έως 85** βαθμούς.

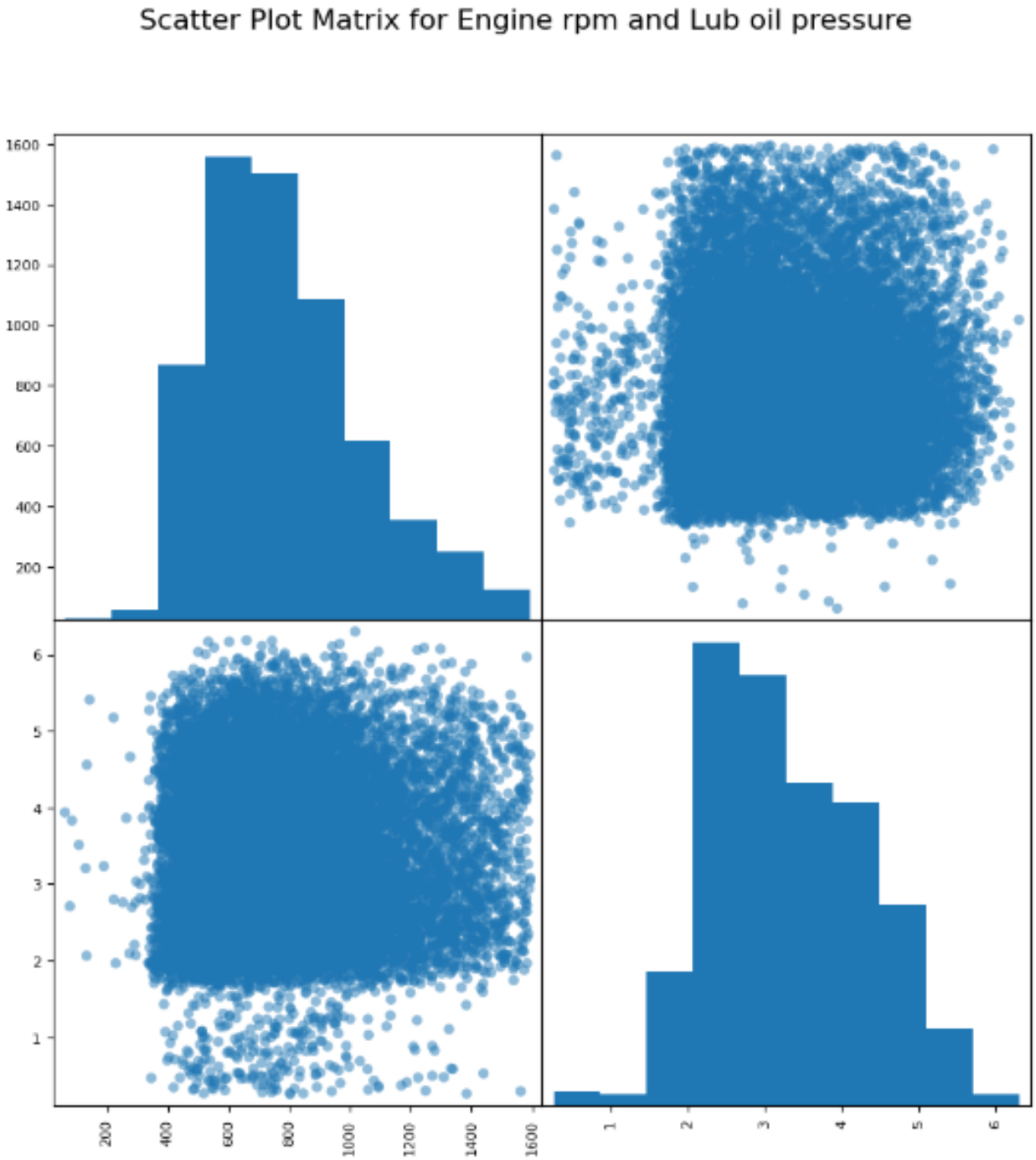
- **Μέση γραμμή (Median):** Η γραμμή μέσα στο κουτί δείχνει την τιμή της διάμεσης **(50ο εκατοστημόριο)**, η οποία φαίνεται να είναι περίπου **80** βαθμοί.
- **Γραμμες (Whiskers):**Οι γραμμές που εκτείνονται από το πλαίσιο προς τα επάνω και προς τα κάτω αντιπροσωπεύουν την εύρεση της μικρότερης και μεγαλύτερης τιμής που δεν είναι ακραίες **(outliers)**. Τα **whiskers** εκτείνονται από περίπου **72,5 βαθμούς έως 87,5 βαθμούς**.
- **Ακραίες Τιμές (Outliers):**Δεν φαίνεται να υπάρχουν ακραίες τιμές εκτός των **whiskers** σε αυτό το διάγραμμα, κάτι που υποδηλώνει ότι όλες οι τιμές πέφτουν εντός των αναμενόμενων όριων για αυτό το σετ δεδομένων.



Σχήμα 4.10: Box plot coolant temp

4.3.5 Πίνακας Διασποράς:

Η δημιουργία **Scatter Plot Matrix** δημιουργεί ένα **scatter plot matrix** για τις επιλεγμένες στήλες **Engine rpm και Lub oil pressure** για να εξεταστούν οι σχέσεις μεταξύ αυτών των χαρακτηριστικών. Το **scatter plot matrix** είναι χρήσιμο για την οπτικοποίηση των αλληλεπιδράσεων μεταξύ πολλαπλών χαρακτηριστικών. Προσφέρει μια συνολική επισκόπηση των σχέσεων μεταξύ των χαρακτηριστικών, επιτρέποντας την ανίχνευση τάσεων και μοτίβων.



Σχήμα 4.11: Scatter plot

Το διάγραμμα παραπάνω είναι ένα **scatter plot matrix** μεταξύ των μεταβλητών **"Engine rpm" και "Lub oil pressure"** με τις αντίστοιχες κατανομές τους παρουσιαζόμενες ως ιστογράμματα στις διαγωνίους θέσεις. Παρουσιάζεται πληροφορία όπως ακολουθεί:

1. **Άξονας Αριστερά-Πάνω (Histogram των Engine rpm):**Το ιστόγραμμα δείχνει την κατανομή των στροφών κινητήρα. Φαίνεται να έχει μια σχετικά κανονική κατανομή με μια συγκέντρωση τιμών γύρω στις **500 έως 1500** στροφές ανά λεπτό.

2. **Άξονας Δεξιά-Κάτω (Histogram των Lub oil pressure):**Η κατανομή της πίεσης λαδιού δείχνει μια σκαλωτή κατανομή με μια εμφανής συγκέντρωση τιμών μεταξύ **2 και 5**.

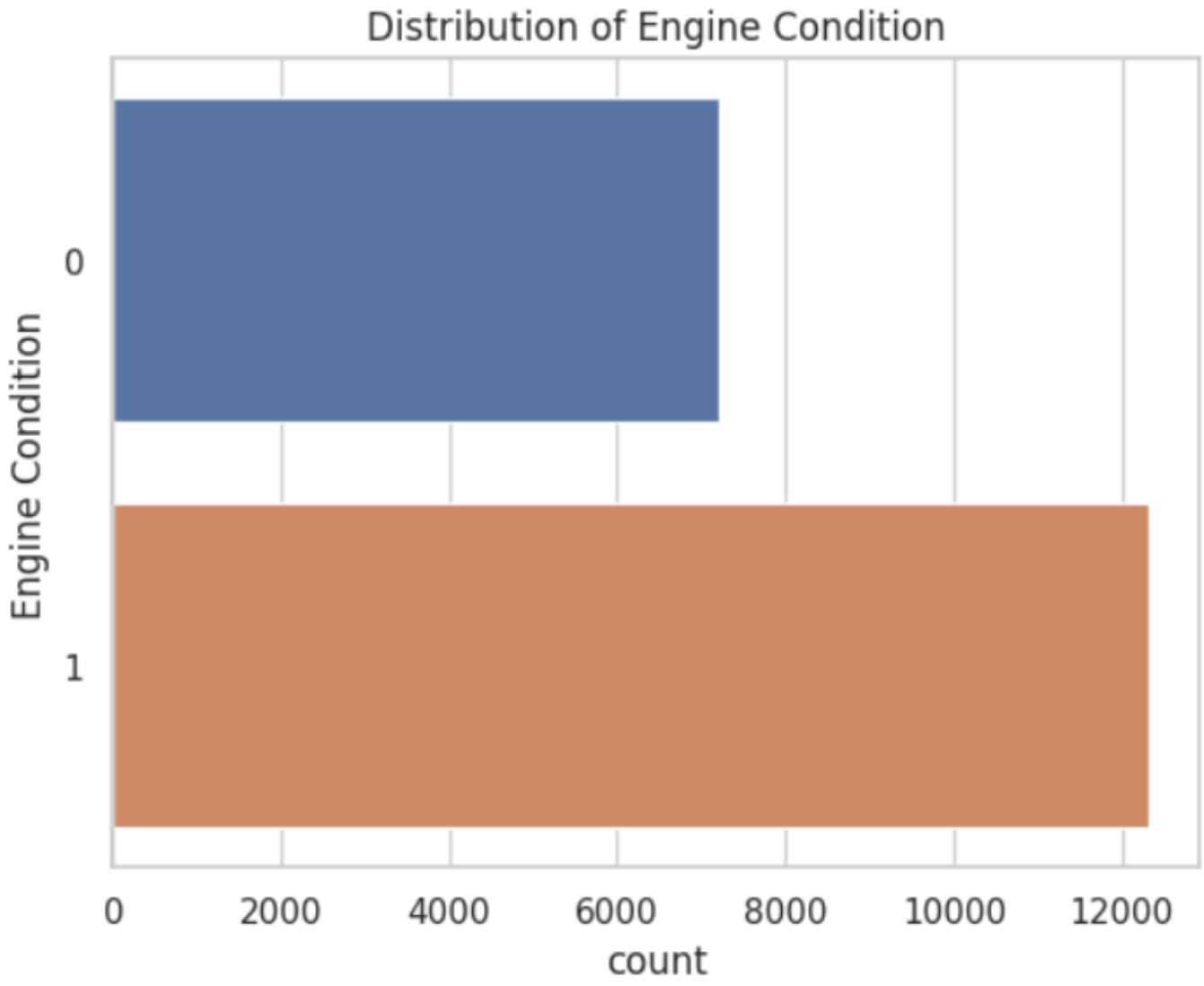
3. **Διαγράμματα Scatter (Μέσα):**

    - Το διάγραμμα στην κορυφή δεξιά δείχνει την σχέση μεταξύ των **"Engine rpm" και "Lub oil pressure"**. Δεν φαίνεται να υπάρχει σαφής γραμμική σχέση μεταξύ των δύο μεταβλητών καθώς τα δεδομένα είναι ευρέως διασκορπισμένα.
    - Το διάγραμμα στο κάτω αριστερά είναι το αντίστροφο του προηγούμενου, δείχνοντας τις ίδιες δεδομενικές τιμές αλλά με τους άξονες να έχουν αλλάξει θέσεις.
    - Τα διαγράμματα αυτά βοηθούν στην ανάλυση της σχέσης και της κατανομής μεταξύ των μεταβλητών, αλλά υποδεικνύουν επίσης την έλλειψη σαφούς γραμμικής σχέσης μεταξύ τους.

4.3.6 Ανάλυση Ανισορροπίας Κλάσεων

1. **Αξιολόγηση ανισορροπίας:**Η κατανομή της μεταβλητής στόχου **Engine Condition** εξετάστηκε για να προσδιοριστεί η ισορροπία των κλάσεων. Διαπιστώθηκε ότι υπάρχει ανισορροπία στις κλάσεις, με την μία κλάση να αποτελεί το **63%** των δειγμάτων και την άλλη το **36%**. Αυτό υποδεικνύει ότι μπορεί να χρειαστεί να εφαρμοστούν τεχνικές όπως το **oversampling ή undersampling** για βελτίωση της εκπαίδευσης των μοντέλων.

2. **Οπτικοποίηση κατανομής κλάσεων με countplot:** Δείχνει την κατανομή των κλάσεων στον στόχο **Engine Condition,** παρέχοντας έναν οπτικό τρόπο για να δούμε την ανισορροπία.



Σχήμα 4.12: Distribution engine condition

5. ΠΛΗΡΟΦΟΡΙΕΣ ΣΥΝΟΛΩΝ ΔΕΔΟΜΕΝΩΝ

5.1 Β) Για την αναγνώριση σημάτων:

Στον κώδικα που αναπτύχθηκε για την εφαρμογή αναγνώρισης τροχαίων σημάτων, η έλλειψη εξειδικευμένων διαδικασιών για την καταγραφή στηλών και περιεχομένων, επιλογή στηλών για ανάλυση και διαδικασίας επιλογής συνδέεται άμεσα με τη φύση των δεδομένων και τον σκοπό του συστήματος. Ο κώδικας εστιάζει κυρίως στην επεξεργασία και αναγνώριση εικόνων παρά στην ανάλυση δομημένων δεδομένων, όπως εκείνα που συναντώνται σε πίνακες ή βάσεις δεδομένων. Οι πληροφορίες που χρειάζονται από τα δεδομένα είναι ουσιαστικά οι εικόνες και οι κατηγορίες τους, όχι επιπλέον μεταδεδομένα ή περιγραφικές πληροφορίες που συνήθως συνοδεύουν τα δεδομένα σε βάσεις δεδομένων.

1. **Φύση Δεδομένων:**Οι εικόνες δεν απαιτούν την καταγραφή στηλών ή την επιλογή ανάμεσα σε διάφορα χαρακτηριστικά **(features)** όπως θα έκανε κάποιος με δομημένα δεδομένα. Η κάθε εικόνα επεξεργάζεται ολόκληρη, με στόχο την εξαγωγή χαρακτηριστικών μέσω του νευρωνικού δικτύου.
2. **Αποτελεσματικότητα Συστήματος:** Η επεξεργασία εστιάζεται στην προετοιμασία των εικόνων για οπτική αναγνώριση, η οποία δεν απαιτεί πολύπλοκες αναλύσεις δεδομένων. Η ανάλυση περιορίζεται στην εφαρμογή αλγορίθμων μηχανικής μάθησης για την κατηγοριοποίηση βάσει των προκύπτοντων χαρακτηριστικών των εικόνων.
3. **Στόχος του Έργου:** Ο στόχος είναι η αναγνώριση και κατηγοριοποίηση τροχαίων σημάτων, και όχι η ερμηνεία ή ανάλυση δεδομένων που θα απαιτούσε την εξέταση και επιλογή διαφορετικών στοιχείων σε έναν πίνακα ή μία βάση δεδομένων.

6. ΒΑΣΙΚΑ ΒΗΜΑΤΑ ΑΝΑΛΥΣΗΣ ΜΟΝΤΕΛΑ (ΑΡΧΙΤΕΚΤΟΝΙΚΗ)

6.1 Α) Για τον κινητήρα MLP :

Σε αυτή την ενότητα θα αναλύσουμε διεξοδικά τα βήματα ανάλυσης και τη δομή των μοντέλων που χρησιμοποιήθηκαν για την πρόβλεψη της κατάστασης του κινητήρα, χρησιμοποιώντας Πολυεπίπεδα Νευρωνικά Δίκτυα **(MLP)**. Η περιγραφή θα είναι εκτενής και λεπτομερής, για να καλύψει πλήρως τα στάδια της διαδικασίας.

6.2 Εισαγωγή στον MLP:

Η διαδικασία ανάλυσης και η ανάπτυξη μοντέλων για την πρόβλεψη της κατάστασης του κινητήρα περιλαμβάνει διάφορα στάδια. Αυτά τα στάδια περιλαμβάνουν τη φόρτωση και την προετοιμασία των δεδομένων, την υπερδειγματοληψία, τον διαχωρισμό των δεδομένων, την κανονικοποίηση, την αρχική εκπαίδευση του **MLP**, την αξιολόγηση της απόδοσης και τη βελτιστοποίηση των υπερπαραμέτρων. Κάθε βήμα είναι κρίσιμο για τη δημιουργία ενός αξιόπιστου και αποδοτικού μοντέλου.

6.3 Φόρτωση και Προετοιμασία Δεδομένων

- **Περιγραφή:**Η αρχική φάση της ανάλυσης ξεκινά με τη φόρτωση των δεδομένων. Τα δεδομένα προέρχονται από ένα αρχείο **CSV** που περιέχει πληροφορίες για τις διάφορες μεταβλητές που επηρεάζουν την κατάσταση του κινητήρα. Το αρχείο **CSV** περιέχει γραμμές δεδομένων, όπου κάθε γραμμή αντιπροσωπεύει μια παρατήρηση και κάθε στήλη αντιπροσωπεύει μια διαφορετική μεταβλητή.
- **Λεπτομέρειες:**Χρησιμοποιούμε τη βιβλιοθήκη **pandas** για να διαβάσουμε το αρχείο **CSV**. Η εντολή **pd.read_csv('engine_data.csv')** φορτώνει τα δεδομένα σε ένα **DataFrame**. Στη συνέχεια, διαχωρίζουμε τα δεδομένα σε δύο κατηγορίες: τα χαρακτηριστικά **(features)** και τον στόχο **(target)**. Τα χαρακτηριστικά είναι οι ανεξάρτητες μεταβλητές που χρησιμοποιούμε για την πρόβλεψη, ενώ ο στόχος είναι η εξαρτημένη μεταβλητή που θέλουμε να προβλέψουμε, η οποία στην προκειμένη περίπτωση είναι η **"κατάσταση του κινητήρα"**.

6.4 Υπερδειγματοληψία (Oversampling)

- **Περιγραφή:**Προκειμένου να αντιμετωπίσουμε την ανισορροπία στις κατηγορίες των δεδομένων, εφαρμόζουμε τη μέθοδο της υπερδειγματοληψίας. Αυτή η τεχνική δημιουργεί νέα δείγματα από την υποεκπροσωπούμενη κατηγορία για να εξισορροπήσει την κατανομή των δεδομένων.
- **Λεπτομέρειες:**Χρησιμοποιούμε τη μέθοδο **RandomOverSampler** από τη βιβλιοθήκη **imblearn** για να πραγματοποιήσουμε την υπερδειγματοληψία. Εφαρμόζουμε την υπερδειγματοληψία στα χαρακτηριστικά και στον στόχο για να δημιουργήσουμε ένα νέο σύνολο δεδομένων με ισορροπημένες κατηγορίες.

6.5 Διαχωρισμός Δεδομένων

- **Περιγραφή:**Αφού έχουμε ισορροπήσει τα δεδομένα, τα διαχωρίζουμε σε εκπαιδευτικό και δοκιμαστικό σύνολο. Το εκπαιδευτικό σύνολο χρησιμοποιείται για την εκπαίδευση του μοντέλου, ενώ το δοκιμαστικό σύνολο χρησιμοποιείται για την αξιολόγηση της απόδοσής του.
- **Λεπτομέρειες:**Χρησιμοποιούμε τη μέθοδο **train_test_split** από τη βιβλιοθήκη **sklearn** για να διαχωρίσουμε τα δεδομένα. Τα δεδομένα χωρίζονται με αναλογία **70%** για το εκπαιδευτικό σύνολο και **30%** για το δοκιμαστικό σύνολο. Ο διαχωρισμός γίνεται τυχαία, αλλά διατηρείται η αναλογία των κατηγοριών με τη χρήση της παραμέτρου **stratify**.

6.6 Κανονικοποίηση (Standard Scaling)

- **Περιγραφή:**Για να διασφαλίσουμε ότι τα δεδομένα μας έχουν την ίδια κλίμακα, εφαρμόζουμε την τεχνική της κανονικοποίησης. Αυτή η διαδικασία είναι σημαντική για την ομαλή λειτουργία των αλγορίθμων μηχανικής μάθησης.
- **Λεπτομέρειες:**Χρησιμοποιούμε το εργαλείο **StandardScaler** από τη βιβλιοθήκη **sklearn** για να κανονικοποιήσουμε τα δεδομένα. Η κανονικοποίηση μετατρέπει τα δεδομένα έτσι ώστε να έχουν μέση τιμή **0** και τυπική απόκλιση **1**. Τα δεδομένα κανονικοποιούνται ξεχωριστά για το εκπαιδευτικό και το δοκιμαστικό σύνολο, με βάση τις τιμές του εκπαιδευτικού συνόλου.

6.7 Αρχική Εκπαίδευση MLP Classifier

- **Περιγραφή:**Η πρώτη εκπαίδευση του Πολυεπίπεδου Νευρωνικού Δικτύου **(MLP)** γίνεται με τις προεπιλεγμένες παραμέτρους **(out-of-the-box)**. Αυτή η αρχική εκπαίδευση βοηθά στην αξιολόγηση της βασικής απόδοσης του μοντέλου πριν από τη βελτιστοποίηση των υπερπαραμέτρων.
- **Λεπτομέρειες:**Το **MLP** είναι ένα είδος νευρωνικού δικτύου που αποτελείται από πολλαπλά επίπεδα νευρώνων. Το δίκτυο περιλαμβάνει ένα είσοδο, ένα ή περισσότερα κρυφά επίπεδα και ένα επίπεδο εξόδου. Εκπαιδεύουμε το **MLP** χρησιμοποιώντας τη μέθοδο **fit** με τα κανονικοποιημένα δεδομένα εκπαίδευσης. Μετά την εκπαίδευση, το μοντέλο πραγματοποιεί προβλέψεις στο δοκιμαστικό σύνολο χρησιμοποιώντας τη μέθοδο **predict**.

6.8 Αξιολόγηση Αρχικής Εκπαίδευσης

- **Περιγραφή:**Μετά την αρχική εκπαίδευση του **MLP**, αξιολογούμε την απόδοσή του χρησιμοποιώντας διάφορα εργαλεία αξιολόγησης. Αυτά τα εργαλεία μας βοηθούν να κατανοήσουμε πόσο καλά λειτουργεί το μοντέλο και πού υπάρχουν περιθώρια βελτίωσης.

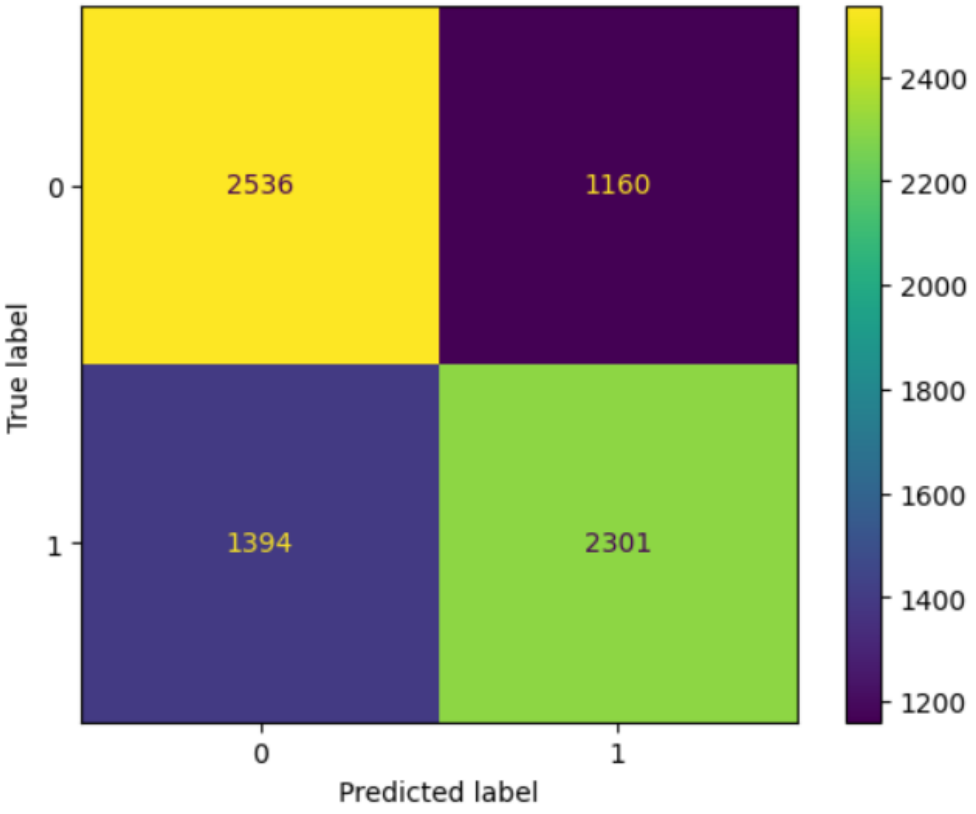
- **Λεπτομέρειες:** Χρησιμοποιούμε τον πίνακα σύγχυσης **(confusion matrix)** για να δούμε πόσα σωστά και λάθος ταξινομημένα δείγματα υπάρχουν για κάθε κατηγορία. Εμφανίζουμε τα αποτελέσματα του πίνακα σύγχυσης γραφικά χρησιμοποιώντας το εργαλείο **ConfusionMatrixDisplay**. Επίσης, χρησιμοποιούμε το εργαλείο **classification_report** για να πάρουμε μια λεπτομερή αναφορά που περιλαμβάνει μετρικές όπως η ακρίβεια, η ανάκληση και το **F1-score** για κάθε κατηγορία.

- **Αναλυτική Επεξήγηση:**

  1. **Εισαγωγή του MLPClassifier:**Χρησιμοποιούμε την κλάση **MLPClassifier** από τη βιβλιοθήκη **sklearn.neural_network** για να δημιουργήσουμε ένα Πολυεπίπεδο Νευρωνικό Δίκτυο.
  2. **Εκπαίδευση του MLPClassifier:**Η μέθοδος **fit** εκπαιδεύει το μοντέλο χρησιμοποιώντας τα κανονικοποιημένα δεδομένα εκπαίδευσης **(X_train_scaled)** και τους αντίστοιχους στόχους **(y_train)**.
  3. **Πρόβλεψη με το MLPClassifier:** Η μέθοδος **predict** χρησιμοποιείται για να προβλέψει τις κατηγορίες των δειγμάτων του δοκιμαστικού συνόλου **(X_test_scaled)**.

- **Αξιολόγηση του Μοντέλου:**

  1. **Πίνακας Σύγχυσης:** Δημιουργούμε τον πίνακα σύγχυσης χρησιμοποιώντας τη συνάρτηση **confusion_matrix**, η οποία συγκρίνει τις πραγματικές κατηγορίες **(y_test)** με τις προβλέψεις **(y_pred_mlp)**.
  2. **Εμφάνιση Πίνακα Σύγχυσης:**Χρησιμοποιούμε την κλάση **ConfusionMatrixDisplay** για να απεικονίσουμε τον πίνακα σύγχυσης και να κατανοήσουμε καλύτερα την απόδοση του μοντέλου σε κάθε κατηγορία.



Σχήμα 6.1: Confusion Matrix MLP

3. **Αναφορά Απόδοσης:** Η συνάρτηση **classification_report** παρέχει μια αναλυτική αναφορά με τις μετρικές απόδοσης όπως η ακρίβεια **(precision)**, η ανάκληση **(recall)** και το **F1-score**.

6.9 Βελτιστοποίηση Υπερπαραμέτρων

- **Περιγραφή:**Μετά την αρχική αξιολόγηση, προχωρούμε στη βελτιστοποίηση των υπερπαραμέτρων του **MLP** για να βελτιώσουμε την απόδοσή του. Οι υπερπαράμετροι είναι ρυθμίσεις του μοντέλου που δεν μαθαίνονται από τα δεδομένα αλλά καθορίζονται πριν από την εκπαίδευση.
- **Λεπτομέρειες:** Χρησιμοποιούμε την τεχνική **GridSearchCV** από τη βιβλιοθήκη **sklearn.model_selection** για να δοκιμάσουμε διάφορους συνδυασμούς υπερπαραμέτρων και να βρούμε τον καλύτερο. **Οι υπερπαράμετροι που βελτιστοποιούμε περιλαμβάνουν:**
    1. **hidden_layer_sizes:**Το μέγεθος και ο αριθμός των κρυφών επιπέδων στο νευρωνικό δίκτυο.
    2. **activation:** Η συνάρτηση ενεργοποίησης που χρησιμοποιείται σε κάθε νευρώνα.
    3. **solver:**Ο αλγόριθμος βελτιστοποίησης που χρησιμοποιείται για την εκπαίδευση του δικτύου.
    4. **alpha:**Η παράμετρος κανονικοποίησης που βοηθά στην αποφυγή υπερπροσαρμογής.
- **Αναλυτική Επεξήγηση : Καθορισμός Υπερπαραμέτρων**
    1. Δημιουργούμε ένα λεξικό **param_dict**που περιέχει τις διάφορες τιμές των υπερπαραμέτρων που θέλουμε να δοκιμάσουμε.
    2. Το **hidden_layer_sizes** καθορίζει τη δομή του νευρωνικού δικτύου, δηλαδή πόσα κρυφά επίπεδα θα υπάρχουν και πόσοι νευρώνες θα έχει το κάθε επίπεδο. Δοκιμάζουμε συνδυασμούς με έναν κρυφό επίπεδο (μεταξύ **1 και 9** νευρώνων) και δύο κρυφά επίπεδα (μεταξύ **1 και 9** νευρώνων στο καθένα).
    3. Για τη συνάρτηση ενεργοποίησης **(activation)**, χρησιμοποιούμε την **'relu'** επειδή είναι γνωστή για την αποδοτικότητά της στην εκπαίδευση βαθιών νευρωνικών δικτύων. Η **'relu'** βοηθά στην αποφυγή του προβλήματος της εξαφάνισης του **gradient**.
    4. Για τον αλγόριθμο βελτιστοποίησης **(solver)**, επιλέγουμε τον **'lbfgs'** επειδή είναι ένας αλγόριθμος βελτιστοποίησης που μπορεί να είναι πιο γρήγορος και αποδοτικός για μικρά σύνολα δεδομένων σε σύγκριση με άλλους αλγόριθμους όπως ο **'adam'** ή ο **'sgd'**.
    5. Η παράμετρος κανονικοποίησης **(alpha)** βοηθά στην αποφυγή υπερπροσαρμογής **(overfitting)** του μοντέλου στα δεδομένα εκπαίδευσης. Δοκιμάζουμε διαφορετικές τιμές της **alpha (0.0001, 0.001, 0.01, 0.1)** για να βρούμε την καλύτερη ισορροπία μεταξύ προσαρμογής και γενίκευσης.
- **Δημιουργία Pipeline:**Δημιουργούμε ένα **pipeline** που περιλαμβάνει το **MLPClassifier**. Το **pipeline** επιτρέπει την ομαλή διαχείριση της διαδικασίας εκπαίδευσης και βελτιστοποίησης.

- **Χρήση GridSearchCV:**Η συνάρτηση **GridSearchCV** δοκιμάζει κάθε συνδυασμό υπερπαραμέτρων από το **param_dict** και εκπαιδεύει ένα μοντέλο για κάθε συνδυασμό. Η διαδικασία αυτή εκτελείται με 5-πλή διασταυρούμενη επικύρωση **(5-fold cross-validation)** για να διασφαλιστεί η αξιοπιστία των αποτελεσμάτων.
- **Εκπαίδευση και Πρόβλεψη:**Το **GridSearchCV** εκπαιδεύει το μοντέλο με τον καλύτερο συνδυασμό υπερπαραμέτρων και πραγματοποιεί προβλέψεις στο δοκιμαστικό σύνολο.
- **Αξιολόγηση Βελτιστοποιημένου Μοντέλου:** Τα αποτελέσματα της πρόβλεψης συγκρίνονται με τις πραγματικές τιμές του συνόλου δοκιμής, και εμφανίζονται αναλυτικά οι μετρικές απόδοσης του μοντέλου (όπως **precision, recall, και f1-score)** μέσω της μεθόδου **classification_report**.
- **Τελική Εμφάνιση Καλύτερου Εκτιμητή:**Μετά την ολοκλήρωση της διαδικασίας βελτιστοποίησης, το **GridSearchCV** παρέχει τον καλύτερο εκτιμητή, δηλαδή το μοντέλο με τον συνδυασμό υπερπαραμέτρων που πέτυχε την καλύτερη απόδοση.

Αυτή η διαδικασία βελτιστοποίησης και αξιολόγησης διασφαλίζει ότι το τελικό μοντέλο **MLP** είναι όσο το δυνατόν πιο αποδοτικό και ακριβές για την πρόβλεψη της κατάστασης του κινητήρα. Μέσω της **GridSearchCV**, εξετάζουμε διάφορους συνδυασμούς υπερπαραμέτρων για να βρούμε τον καλύτερο, ενώ η χρήση του **pipeline** καθιστά τη διαδικασία πιο ομαλή και διαχειρίσιμη.

7. ΒΑΣΙΚΑ ΒΗΜΑΤΑ ΑΝΑΛΥΣΗΣ & ΜΟΝΤΕΛΑ (ΑΡΧΙΤΕΚΤΟΝΙΚΗ)

7.1 Α) Για τον κινητήρα SVM :

Σε αυτή την ενότητα, θα αναλύσουμε λεπτομερώς τα βήματα ανάλυσης και τη δομή των μοντέλων που χρησιμοποιήθηκαν για την πρόβλεψη της κατάστασης του κινητήρα, χρησιμοποιώντας Υποστηρικτικούς Διανυσματικούς Μηχανισμούς **(Support Vector Machines - SVM)**. Η περιγραφή θα είναι εκτενής και λεπτομερής, καλύπτοντας όλα τα στάδια της διαδικασίας.

7.2 Εισαγωγή στο SVM :

Οι Υποστηρικτικοί Διανυσματικοί Μηχανισμοί **(SVM)** είναι ένας από τους πιο δημοφιλείς και ισχυρούς αλγορίθμους μηχανικής μάθησης για ταξινόμηση και παλινδρόμηση. Το **SVM** λειτουργεί βρίσκοντας το υπερεπίπεδο που διαχωρίζει καλύτερα τις κατηγορίες των δεδομένων σε έναν υψηλής διάστασης χώρο.

7.3 Φόρτωση και Προετοιμασία Δεδομένων

- **Περιγραφή:**Η αρχική φάση της ανάλυσης ξεκινά με τη φόρτωση των δεδομένων. Τα δεδομένα προέρχονται από ένα αρχείο **CSV** που περιέχει πληροφορίες για τις διάφορες μεταβλητές που επηρεάζουν την κατάσταση του κινητήρα. Το αρχείο **CSV** περιέχει γραμμές δεδομένων, όπου κάθε γραμμή αντιπροσωπεύει μια παρατήρηση και κάθε στήλη αντιπροσωπεύει μια διαφορετική μεταβλητή.
- **Λεπτομέρειες:**Χρησιμοποιούμε τη βιβλιοθήκη **pandas** για να διαβάσουμε το αρχείο **CSV**. Η εντολή **pd.read_csv('engine_data.csv')** φορτώνει τα δεδομένα σε ένα **DataFrame**. Στη συνέχεια, διαχωρίζουμε τα δεδομένα σε δύο κατηγορίες: τα χαρακτηριστικά **(features)** και τον στόχο **(target)**. Τα χαρακτηριστικά είναι οι ανεξάρτητες μεταβλητές που χρησιμοποιούμε για την πρόβλεψη, ενώ ο στόχος είναι η εξαρτημένη μεταβλητή που θέλουμε να προβλέψουμε, η οποία στην προκειμένη περίπτωση είναι η **"κατάσταση του κινητήρα"**.

7.4 Υπερδειγματοληψία (Oversampling)

- **Περιγραφή:**Για να αντιμετωπίσουμε την ανισορροπία στις κατηγορίες των δεδομένων, εφαρμόζουμε τη μέθοδο της υπερδειγματοληψίας. Αυτή η τεχνική δημιουργεί νέα δείγματα από την υποεκπροσωπούμενη κατηγορία για να εξισορροπήσει την κατανομή των δεδομένων.
- **Λεπτομέρειες:**Χρησιμοποιούμε τη μέθοδο **RandomOverSampler** από τη βιβλιοθήκη **imblearn** για να πραγματοποιήσουμε την υπερδειγματοληψία.

  Εφαρμόζουμε την υπερδειγματοληψία στα χαρακτηριστικά και στον στόχο για να δημιουργήσουμε ένα νέο σύνολο δεδομένων με ισορροπημένες κατηγορίες.

7.5 Διαχωρισμός Δεδομένων

- **Περιγραφή:**Αφού έχουμε ισορροπήσει τα δεδομένα, τα διαχωρίζουμε σε εκπαιδευτικό και δοκιμαστικό σύνολο. Το εκπαιδευτικό σύνολο χρησιμοποιείται για την εκπαίδευση του μοντέλου, ενώ το δοκιμαστικό σύνολο χρησιμοποιείται για την αξιολόγηση της απόδοσής του.
- **Λεπτομέρειες:**Χρησιμοποιούμε τη μέθοδο **train_test_split** από τη βιβλιοθήκη **sklearn** για να διαχωρίσουμε τα δεδομένα. Τα δεδομένα χωρίζονται με αναλογία **70%** για το εκπαιδευτικό σύνολο και **30%** για το δοκιμαστικό σύνολο. Ο διαχωρισμός γίνεται τυχαία, αλλά διατηρείται η αναλογία των κατηγοριών με τη χρήση της παραμέτρου **stratify**.

7.6 Κανονικοποίηση (Standard Scaling)

- **Περιγραφή:**Για να διασφαλίσουμε ότι τα δεδομένα μας έχουν την ίδια κλίμακα, εφαρμόζουμε την τεχνική της κανονικοποίησης. Αυτή η διαδικασία είναι σημαντική για την ομαλή λειτουργία των αλγορίθμων μηχανικής μάθησης.
- **Λεπτομέρειες:**Χρησιμοποιούμε το εργαλείο **StandardScaler** από τη βιβλιοθήκη **sklearn** για να κανονικοποιήσουμε τα δεδομένα. Η κανονικοποίηση μετατρέπει τα δεδομένα έτσι ώστε να έχουν μέση τιμή **0**και τυπική απόκλιση **1**. Τα δεδομένα κανονικοποιούνται ξεχωριστά για το εκπαιδευτικό και το δοκιμαστικό σύνολο, με βάση τις τιμές του εκπαιδευτικού συνόλου.

7.7 Αρχική Εκπαίδευση SVM Classifier

- **Περιγραφή:**Η πρώτη εκπαίδευση του Υποστηρικτικού Διανυσματικού Μηχανισμού **(SVM)** γίνεται με τις προεπιλεγμένες παραμέτρους **(out-of-the-box)**. Αυτή η αρχική εκπαίδευση βοηθά στην αξιολόγηση της βασικής απόδοσης του μοντέλου πριν από τη βελτιστοποίηση των υπερπαραμέτρων.
- **Λεπτομέρειες:**Το **SVM** είναι ένας ισχυρός αλγόριθμος μηχανικής μάθησης που χρησιμοποιείται για ταξινόμηση και παλινδρόμηση. Το **SVM** βρίσκει το υπερεπίπεδο που διαχωρίζει καλύτερα τις κατηγορίες των δεδομένων. Εκπαιδεύουμε το **SVM** χρησιμοποιώντας τη μέθοδο **fit** με τα κανονικοποιημένα δεδομένα εκπαίδευσης. Μετά την εκπαίδευση, το μοντέλο πραγματοποιεί προβλέψεις στο δοκιμαστικό σύνολο χρησιμοποιώντας τη μέθοδο **predict**.

7.8 Αξιολόγηση Αρχικής Εκπαίδευσης

- **Περιγραφή:**Μετά την αρχική εκπαίδευση του **SVM**, αξιολογούμε την απόδοσή του χρησιμοποιώντας διάφορα εργαλεία αξιολόγησης. Αυτά τα εργαλεία μας βοηθούν να κατανοήσουμε πόσο καλά λειτουργεί το μοντέλο και πού υπάρχουν περιθώρια βελτίωσης.

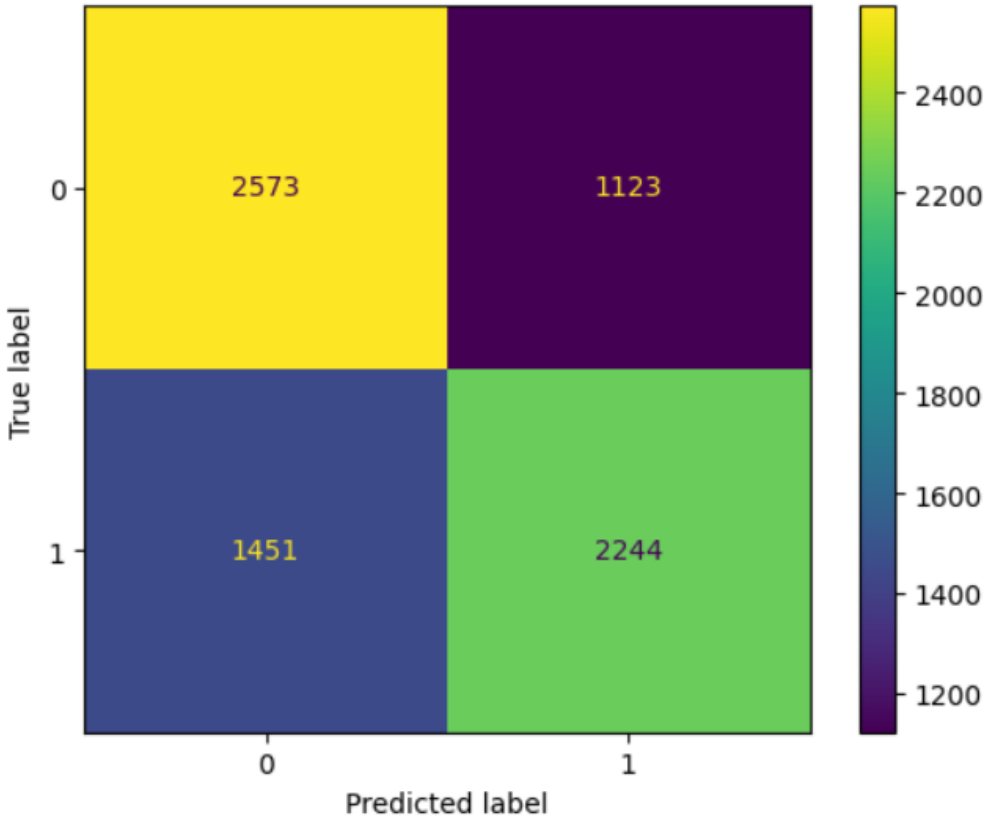
- **Λεπτομέρειες:**Χρησιμοποιούμε τον πίνακα σύγχυσης **(confusion matrix)** για να δούμε πόσα σωστά και λάθος ταξινομημένα δείγματα υπάρχουν για κάθε κατηγορία. Εμφανίζουμε τα αποτελέσματα του πίνακα σύγχυσης γραφικά χρησιμοποιώντας το εργαλείο **ConfusionMatrixDisplay**. Επίσης, χρησιμοποιούμε το εργαλείο **classification_report** για να πάρουμε μια λεπτομερή αναφορά που περιλαμβάνει μετρικές όπως η ακρίβεια, η ανάκληση και το **F1-score** για κάθε κατηγορία.

- **Αναλυτική Επεξήγηση:**

  1. **Εισαγωγή του SVM Classifier:** Χρησιμοποιούμε την κλάση **SVC** από τη βιβλιοθήκη **sklearn.svm** για να δημιουργήσουμε ένα Υποστηρικτικό Διανυσματικό Μηχανισμό.
  2. **Εκπαίδευση του SVM Classifier:** Η μέθοδος **fit** εκπαιδεύει το μοντέλο χρησιμοποιώντας τα κανονικοποιημένα δεδομένα εκπαίδευσης **(X_train_scaled)** και τους αντίστοιχους στόχους **(y_train)**.
  3. **Πρόβλεψη με το SVM Classifier:** Η μέθοδος **predict** χρησιμοποιείται για να προβλέψει τις κατηγορίες των δειγμάτων του δοκιμαστικού συνόλου **(X_test_scaled)**.

- **Αξιολόγηση του Μοντέλου:**

  1. **Πίνακας Σύγχυσης:** Δημιουργούμε τον πίνακα σύγχυσης χρησιμοποιώντας τη συνάρτηση **confusion_matrix**, η οποία συγκρίνει τις πραγματικές κατηγορίες **(y_test)** με τις προβλέψεις **(y_pred_svc)**.
  2. **Εμφάνιση Πίνακα Σύγχυσης:** Χρησιμοποιούμε την κλάση **ConfusionMatrixDisplay** για να απεικονίσουμε τον πίνακα σύγχυσης και να κατανοήσουμε καλύτερα την απόδοση του μοντέλου σε κάθε κατηγορία.



Σχήμα 7.1: Confusion Matrix SVM

3. **Αναφορά Απόδοσης:** Η συνάρτηση **classification_report** παρέχει μια αναλυτική αναφορά με τις μετρικές απόδοσης όπως η ακρίβεια **(precision)**, η ανάκληση **(recall)** και το **F1-score**.

7.9 Βελτιστοποίηση Υπερπαραμέτρων

- **Περιγραφή:**Μετά την αρχική αξιολόγηση, προχωρούμε στη βελτιστοποίηση των υπερπαραμέτρων του **SVM** για να βελτιώσουμε την απόδοσή του. Οι υπερπαράμετροι είναι ρυθμίσεις του μοντέλου που δεν μαθαίνονται από τα δεδομένα αλλά καθορίζονται πριν από την εκπαίδευση.
- **Λεπτομέρειες:**Χρησιμοποιούμε την τεχνική **GridSearchCV** από τη βιβλιοθήκη **sklearn.model_** για να δοκιμάσουμε διάφορους συνδυασμούς υπερπαραμέτρων και να βρούμε τον καλύτερο. **Οι υπερπαράμετροι που βελτιστοποιούμε περιλαμβάνουν:**
    1. **kernel:** Ο τύπος του πυρήνα που χρησιμοποιείται για τον μετασχηματισμό των δεδομένων. **C:** Η παράμετρος κανονικοποίησης που ελέγχει την επιβολή ποινής για λάθη ταξινόμησης.
    2. **gamma:** Η παράμετρος που ελέγχει την καμπυλότητα του χώρου απόφασης.
    3. **degree:** Ο βαθμός του πολυωνυμικού πυρήνα.
- **Αναλυτική Επεξήγηση Γραμμικού Πυρήνα (Linear Kernel):**
    1. **Παράμετροι:** Για τον γραμμικό πυρήνα, η μοναδική υπερπαράμετρος είναι η σταθερά **C**. Η σταθερά **C** ελέγχει την ισορροπία μεταξύ του μεγέθους του περιθωρίου και του σφάλματος ταξινόμησης.
    2. **Λόγοι Επιλογής:** Η υπερπαράμετρος **C** επιλέγεται επειδή είναι η κύρια παράμετρος που επηρεάζει την απόδοση του **γραμμικού πυρήνα**. Αυξάνοντας την τιμή της **C**, το μοντέλο γίνεται πιο επιρρεπές σε **overfitting**, ενώ μειώνοντας την τιμή της **C**, το μοντέλο γίνεται πιο γενικευμένο.
    3. **Παράμετροι για GridSearchCV:**param_dict_linear = "svm_C": [0.01, 0.1, 1, 10, 100]
- **Αναλυτική Επεξήγηση Πολυωνυμικού Πυρήνα (Polynomial Kernel):**
    1. **Παράμετροι:** Οι υπερπαράμετροι για τον πολυωνυμικό πυρήνα περιλαμβάνουν την σταθερά **C**, το **gamma** και το **degree**. Το **gamma** ελέγχει την καμπυλότητα του χώρου απόφασης, ενώ το **degree** καθορίζει τον βαθμό του πολυωνυμίου.
    2. **Λόγοι Επιλογής:** Οι παράμετροι αυτές επιλέγονται επειδή επηρεάζουν σημαντικά την πολυπλοκότητα και την ευελιξία του **πολυωνυμικού πυρήνα**. Η επιλογή των κατάλληλων τιμών για αυτές τις παραμέτρους είναι κρίσιμη για την απόδοση του μοντέλου.
    3. **Παράμετροι για GridSearchCV:** param_dict_poly = "svm_C": [0.01, 0.1, 1, 10, 100], "svm_gamma": ["scale", "auto"], "svm_degree": [1, 2, 3, 4, 5]

- **Αναλυτική Επεξήγηση Πυρήνα Ακτινικής Βάσης (RBF Kernel):**

    1. **Παράμετροι:** Οι υπερπαράμετροι για τον πυρήνα ακτινικής βάσης περιλαμβάνουν την σταθερά **C** και το **gamma**. Το **gamma**ελέγχει την καμπυλότητα του χώρου απόφασης.
    2. **Λόγοι Επιλογής:** Οι παράμετροι αυτές επιλέγονται επειδή επηρεάζουν σημαντικά την ικανότητα του μοντέλου να διαχωρίζει τα δεδομένα σε έναν μη γραμμικό χώρο. Το **gamma** καθορίζει την ακτίνα επιρροής των δειγμάτων εκπαίδευσης.
    3. **Παράμετροι για GridSearchCV:** param_dict_rbf = "svm_C": [0.01, 0.1, 1, 10, 100], "svm_gamma": ["scale", "auto"]

- **Αναλυτική Επεξήγηση Σιγμοειδή Πυρήνα (Sigmoid Kernel):**

    1. **Παράμετροι:** Οι υπερπαράμετροι για τον σιγμοειδή πυρήνα περιλαμβάνουν την σταθερά **C** και το **gamma**. Το **gamma** ελέγχει την καμπυλότητα του χώρου απόφασης.
    2. **Λόγοι Επιλογής:** Οι παράμετροι αυτές επιλέγονται επειδή επηρεάζουν τη συμπεριφορά του μοντέλου σε μη γραμμικούς χώρους και είναι ιδιαίτερα χρήσιμες για προβλήματα ταξινόμησης με πολύπλοκα σύνορα απόφασης.
    3. **Παράμετροι για GridSearchCV:** param_dict_sigmoid = "svm_C": [0.01, 0.1, 1, 10, 100], "svm_gamma": ["scale", "auto"]

- **Δημιουργία Pipeline:** Δημιουργούμε ένα **Pipeline** που περιλαμβάνει το **SVM Classifier**, επιτρέποντας την εύκολη διαχείριση της διαδικασίας εκπαίδευσης και βελτιστοποίησης.

- **Χρήση GridSearchCV:** Η συνάρτηση **GridSearchCV** δοκιμάζει κάθε συνδυασμό υπερπαραμέτρων από το **param_dict** και εκπαιδεύει ένα μοντέλο για κάθε συνδυασμό. Η διαδικασία αυτή εκτελείται με 5-πλή διασταυρούμενη επικύρωση **(5-fold cross-validation)** για να διασφαλιστεί η αξιοπιστία των αποτελεσμάτων.

- **Εκπαίδευση και Πρόβλεψη:** Το **GridSearchCV** εκπαιδεύει το μοντέλο με τον καλύτερο συνδυασμό υπερπαραμέτρων και πραγματοποιεί προβλέψεις στο δοκιμαστικό σύνολο.

- **Αξιολόγηση Βελτιστοποιημένου Μοντέλου:** Τα αποτελέσματα της πρόβλεψης συγκρίνονται με τις πραγματικές τιμές του συνόλου δοκιμής, και εμφανίζονται αναλυτικά οι μετρικές απόδοσης του μοντέλου (όπως **precision, recall, και f1-score)** μέσω της μεθόδου **classification_report**.

- **Τελική Εμφάνιση Καλύτερου Εκτιμητή:**Μετά την ολοκλήρωση της διαδικασίας βελτιστοποίησης, το **GridSearchCV** παρέχει τον καλύτερο εκτιμητή, δηλαδή το μοντέλο με τον συνδυασμό υπερπαραμέτρων που πέτυχε την καλύτερη απόδοση. Ο καλύτερος εκτιμητής εμφανίζεται χρησιμοποιώντας τη συνάρτηση **best_estimator_** του **GridSearchCV**.

  Αυτή η διαδικασία βελτιστοποίησης και αξιολόγησης διασφαλίζει ότι το τελικό μοντέλο **SVM** είναι όσο το δυνατόν πιο αποδοτικό και ακριβές για την πρόβλεψη της κατάστασης του κινητήρα.

8. ΒΑΣΙΚΑ ΒΗΜΑΤΑ ΑΝΑΛΥΣΗΣ & ΜΟΝΤΕΛΑ (ΑΡΧΙΤΕΚΤΟΝΙΚΗ)

8.1 Β) Για την αναγνώριση σημάτων

8.2 Εισαγωγή :

Σε αυτή την ενότητα θα αναλύσουμε διεξοδικά τα βήματα ανάλυσης και τη δομή του μοντέλου που χρησιμοποιήθηκε για την αναγνώριση πινακίδων κυκλοφορίας, χρησιμοποιώντας συνελικτικά νευρωνικά δίκτυα **(Convolutional Neural Networks - CNN)**. Η περιγραφή θα είναι εκτενής και λεπτομερής, καλύπτοντας πλήρως τα στάδια της διαδικασίας για το **train model** και το **test model**.

Η διαδικασία ανάλυσης και ανάπτυξης του μοντέλου περιλαμβάνει διάφορα στάδια. Αυτά τα στάδια περιλαμβάνουν:

1. Φόρτωση και προετοιμασία των δεδομένων **(Train Model)**.
2. Προεπεξεργασία των εικόνων **(Train & Test Model)**.
3. Διαχωρισμός των δεδομένων **(Train Model)**.
4. Κατασκευή του μοντέλου **(Train Model)**.
5. Συμπλήρωση και εκπαίδευση του μοντέλου **(Train Model)**.
6. Αποθήκευση του τελικού μοντέλου **(Train Model)**.
7. Αξιολόγηση του μοντέλου και Πρόβλεψη και αξιολόγηση με νέα δεδομένα **(Train & Test Model)**.

   Κάθε βήμα είναι κρίσιμο για τη δημιουργία ενός αξιόπιστου και αποδοτικού μοντέλου.

8.3 Φόρτωση και Προετοιμασία Δεδομένων (Train Model)

- **Περιγραφή:**Η αρχική φάση της ανάλυσης ξεκινά με τη φόρτωση των δεδομένων στο **train model**. Τα δεδομένα προέρχονται από ένα σύνολο εικόνων ταξινομημένων σε κατηγορίες που αντιστοιχούν σε διάφορες πινακίδες κυκλοφορίας.
- **Λεπτομέρειες:**
  1. **Χρήση βιβλιοθήκης OpenCV:** Η βιβλιοθήκη **OpenCV** χρησιμοποιείται για την ανάγνωση και επεξεργασία των εικόνων.
  2. **Φόρτωση και ανάλυση υποφακέλων:** Κάθε εικόνα από το σύνολο δεδομένων φορτώνεται, μετατρέπεται σε ασπρόμαυρη και κανονικοποιείται πριν αποθηκευτεί στη λίστα εικόνων και στη λίστα αριθμών κλάσεων.

8.4 Προεπεξεργασία Εικόνων (Train & Test Model)

- **Περιγραφή:**Για την καλύτερη απόδοση του μοντέλου, οι εικόνες προεπεξεργάζονται τόσο στο **train model** όσο και στο **test model** ώστε να είναι ασπρόμαυρες και εξισορροπημένες ως προς τον φωτισμό. Επίσης, οι τιμές τους κανονικοποιούνται στην περιοχή **0-1**.
- **Λεπτομέρειες:**
  1. **Μετατροπή σε ασπρόμαυρο:** Η λειτουργία **grayscale(img)** μετατρέπει την εικόνα σε ασπρόμαυρη με ένα κανάλι χρησιμοποιώντας τη βιβλιοθήκη **OpenCV**
  2. **Εξισορρόπηση φωτισμού:** Η λειτουργία **equalize(img)** εξισορροπεί το φωτισμό στην εικόνα χρησιμοποιώντας τη μέθοδο εξισορρόπησης ιστογράμματος της **OpenCV**.
  3. **Κανονικοποίηση τιμών:** Οι τιμές των εικόνων κανονικοποιούνται από το εύρος **0-255** στο εύρος **0-1**.

8.5 Διαχωρισμός Δεδομένων (Train Model)

- **Περιγραφή:**Αφού έχουμε προεπεξεργαστεί τα δεδομένα, τα διαχωρίζουμε στο **train model** σε εκπαιδευτικό, επικύρωσης και δοκιμαστικό σύνολο. Το εκπαιδευτικό σύνολο χρησιμοποιείται για την εκπαίδευση του μοντέλου, το σύνολο επικύρωσης για την αξιολόγηση κατά τη διάρκεια της εκπαίδευσης και το δοκιμαστικό σύνολο για την τελική αξιολόγηση.
- **Λεπτομέρειες:** Διαχωρίζουμε σε σύνολα και χρησιμοποιούμε τη μέθοδο **train_test_split** από τη βιβλιοθήκη **sklearn** για να διαχωρίσουμε τα δεδομένα σε εκπαιδευτικό, επικύρωσης και δοκιμαστικό σύνολο με αναλογία **60-20-20**.

8.6 Κατασκευή του Μοντέλου (Train Model)

- **Περιγραφή:**Η αρχιτεκτονική του μοντέλου στο **train model** περιλαμβάνει συνελικτικά **(convolutional)**επίπεδα, επίπεδα **max-pooling**, πλήρως συνδεδεμένα **(fully connected)** επίπεδα και επίπεδα κανονικοποίησης **(dropout)**.
- **Λεπτομέρειες:**
  1. **Συνελικτικά επίπεδα:** Το μοντέλο χρησιμοποιεί τρία συνελικτικά επίπεδα για την εξαγωγή χαρακτηριστικών από τις εικόνες.
  2. **Επίπεδο ισοπέδωσης (Flatten):** Μετατρέπει τα δεδομένα από δισδιάστατα πίνακες σε μονοδιάστατο πίνακα.
  3. **Πλήρως συνδεδεμένα επίπεδα:** Το μοντέλο περιλαμβάνει δύο πλήρως συνδεδεμένα επίπεδα με **512 και 256** νευρώνες αντίστοιχα.
  4. **Επίπεδα κανονικοποίησης (Dropout):** Χρησιμοποιούνται για την αποφυγή υπερπροσαρμογής.
  5. **Έξοδος:** Το τελικό επίπεδο είναι πλήρως συνδεδεμένο με αριθμό νευρώνων ίσο με τον αριθμό των κατηγοριών **(12)**, χρησιμοποιώντας την ενεργοποίηση **softmax**.

8.7 Συμπλήρωση και Εκπαίδευση του Μοντέλου (Train Model)

- **Περιγραφή:**Το μοντέλο συμπληρώνεται στο **train model** με τον **optimizer Adam** και το **loss function** κατηγορική διασταυρούμενη εντροπία **(categorical crossentropy)**. Η εκπαίδευση πραγματοποιείται για 15 εποχές **(epochs)** με **batch size 64**.
- **Λεπτομέρειες:**
    1. **Συμπλήρωση του μοντέλου:** Το μοντέλο συμπληρώνεται χρησιμοποιώντας τον **optimizer Adam**, το **loss function** κατηγορική διασταυρούμενη εντροπία και την μέτρηση ακρίβειας.
    2. **Εκπαίδευση:** Το μοντέλο εκπαιδεύεται με τα δεδομένα εκπαίδευσης και επικύρωσης για **15 εποχές**.

8.8 Αποθήκευση του Μοντέλου (Train Model)

- **Περιγραφή:**Μετά την ολοκλήρωση της εκπαίδευσης στο **train model**, το τελικό μοντέλο αποθηκεύεται για μελλοντική χρήση.
- **Λεπτομέρειες:** Το μοντέλο αποθηκεύεται σε αρχείο **H5** χρησιμοποιώντας τη μέθοδο **save της Keras**.

8.9 Αξιολόγηση του Μοντέλου (Train & Test Model)

- **Περιγραφή:**Μετά την εκπαίδευση, το μοντέλο αξιολογείται τόσο στο **train model** όσο και στο **test model** με βάση τα δοκιμαστικά δεδομένα.
- **Λεπτομέρειες:**
    1. **Πρόβλεψη και Αξιολόγηση (Test Model):** Το μοντέλο στο **test model** προβλέπει τις κατηγορίες των δοκιμαστικών δεδομένων και υπολογίζει την ακρίβεια.
    2. **Μετρικές:** Η αξιολόγηση του μοντέλου γίνεται με χρήση των μετρικών **accuracy και loss**. Η ακρίβεια **(accuracy)** μετράει το ποσοστό των σωστών προβλέψεων, ενώ η απώλεια **(loss)** μετράει τη διαφορά μεταξύ των προβλέψεων του μοντέλου και των πραγματικών τιμών.
- **Ανάλυση της Αρχιτεκτονικής του Νευρωνικού Δικτύου και Αναλυτική περιγραφή των επιπέδων του μοντέλου:**
    1. **Conv2D:** Τα συνελικτικά επίπεδα **(Conv2D)** εξάγουν χαρακτηριστικά από τις εικόνες εφαρμόζοντας φίλτρα συνελίκωσης. Αυτό βοηθά στην αναγνώριση μοτίβων και χαρακτηριστικών στις εικόνες
    2. **MaxPooling2D:**Τα επίπεδα **max-pooling** μειώνουν τη διάσταση των χαρακτηριστικών, διατηρώντας τα πιο σημαντικά χαρακτηριστικά και μειώνοντας τον υπολογιστικό φόρτο.
    3. **Flatten:** Το επίπεδο ισοπέδωσης **(Flatten)** μετατρέπει τα δεδομένα από δισδιάστατα πίνακες σε μονοδιάστατο πίνακα, προετοιμάζοντας τα για τα πλήρως συνδεδεμένα επίπεδα.

4. **Dense:** Τα πλήρως συνδεδεμένα επίπεδα **(Dense)** εκτελούν την τελική ταξινόμηση. Χρησιμοποιούν την ενεργοποίηση **ReLU** για μη γραμμικότητα και την ενεργοποίηση **softmax** για την τελική έξοδο.
5. **Dropout:** Τα επίπεδα κανονικοποίησης **(Dropout)** αποτρέπουν την υπερπροσαρμογή, απενεργοποιώντας τυχαία ένα ποσοστό νευρώνων κατά την εκπαίδευση.

- **Εξήγηση της συμβολής κάθε επιπέδου στη συνολική απόδοση του μοντέλου:**

1. **Conv2D και MaxPooling2D:**Αυτά τα επίπεδα βοηθούν στην αναγνώριση και τη μείωση χαρακτηριστικών, καθιστώντας το μοντέλο ικανό να γενικεύει καλά σε νέα δεδομένα.
2. **Flatten και Dense:** Τα επίπεδα αυτά εκτελούν την τελική ταξινόμηση, βασιζόμενα στα χαρακτηριστικά που έχουν εξαχθεί από τα προηγούμενα επίπεδα.
3. **Dropout:** Βελτιώνει την ικανότητα του μοντέλου να γενικεύει, αποτρέποντας την υπερπροσαρμογή.

- **Λεπτομερής Περιγραφή των Συναρτήσεων Προεπεξεργασίας:**

1. **Grayscale:** Μετατρέπει την εικόνα σε ασπρόμαυρη με ένα κανάλι, μειώνοντας την πολυπλοκότητα και το μέγεθος των δεδομένων.
2. **Equalize:** Εξισορροπεί το φωτισμό στην εικόνα, βελτιώνοντας την ποιότητα και την ομοιομορφία των δεδομένων εισόδου.
3. **Κανονικοποίηση τιμών:** Κανονικοποιεί τις τιμές των εικόνων στην περιοχή 0-1, βελτιώνοντας την απόδοση του μοντέλου κατά την εκπαίδευση.

- **Γιατί επιλέχθηκε η συγκεκριμένη προσέγγιση:**

1. **Grayscale:** Απλοποιεί τα δεδομένα, καθιστώντας τα κατάλληλα για την επεξεργασία https://www.overleaf.com/project/663e0efe80dfd5d86f41d368chapter.1από το μοντέλο.
2. **Equalize:** Βελτιώνει την ποιότητα των εικόνων, καθιστώντας τις πιο ομοιογενείς.
3. **Κανονικοποίηση:** Βοηθά στην ταχύτερη και σταθερότερη εκπαίδευση του μοντέλου.

9. ΑΠΟΤΕΛΕΣΜΑΤΑ

9.1 SVM

SVM "Out of the Box" (χωρίς βελτιστοποίηση υπερ παραμέτρων)

	precision	recall	f1-score	support
0	0.64	0.70	0.67	3696
1	0.67	0.61	0.64	3695
accuracy			0.65	7391
macro avg	0.65	0.65	0.65	7391
weighted avg	0.65	0.65	0.65	7391

Σχήμα 9.1: SVM OUT OF THE BOX

1. **Precision, Recall, F1-Score:**
   - **Κλάση 0:** Precision = 0.64, Recall = 0.70, F1-Score = 0.67
   - **Κλάση 1:** Precision = 0.67, Recall = 0.61, F1-Score = 0.64
2. **Overall:**
   - Accuracy = 0.65
   - Macro Avg = 0.65
   - Weighted Avg = 0.65
3. **ΣΥΜΠΕΡΑΣΜΑΤΑ**
   - Η ακρίβεια (accuracy) είναι 0.65, με το μοντέλο να έχει ελαφρώς καλύτερη απόδοση στην κλάση 1 σε σχέση με την κλάση 0.
   - Το μοντέλο έχει ισορροπημένη απόδοση (macro και weighted averages είναι ίσα).

Γραμμικός Πυρήνας (C=125)

	precision	recall	f1-score	support
0	0.66	0.50	0.57	3696
1	0.60	0.75	0.67	3695
accuracy			0.62	7391
macro avg	0.63	0.62	0.62	7391
weighted avg	0.63	0.62	0.62	7391

Σχήμα 9.2: ΓΡΑΜΜΙΚΟΣ ΠΥΡΗΝΑΣ SVM

1. **Precision, Recall, F1-Score:**
   - **Κλάση 0:** Precision = 0.66, Recall = 0.50, F1-Score = 0.57
   - **Κλάση 1:** Precision = 0.60, Recall = 0.75, F1-Score = 0.67
2. **Overall:**
   - Accuracy = 0.62
   - Macro Avg = 0.63
   - Weighted Avg = 0.62
3. **ΣΥΜΠΕΡΑΣΜΑΤΑ**
   - Η ακρίβεια (accuracy) μειώνεται σε 0.62.
   - Η κλάση 0 έχει καλύτερη precision αλλά χαμηλότερη recall, ενώ η κλάση 1 έχει το αντίθετο. Αυτό σημαίνει ότι το μοντέλο κάνει περισσότερα λάθη κατά την πρόβλεψη της κλάσης 0.

Πολυωνυμικός Πυρήνας (C=100, degree=1, kernel='poly')

	precision	recall	f1-score	support
0	0.65	0.59	0.62	3696
1	0.62	0.68	0.65	3695
accuracy			0.63	7391
macro avg	0.63	0.63	0.63	7391
weighted avg	0.63	0.63	0.63	7391

Σχήμα 9.3: ΠΟΛΥΩΝΥΜΙΚΟΣ ΠΥΡΗΝΑΣ SVM

1. **Precision, Recall, F1-Score:**
   - **Κλάση 0:**Precision = 0.65, Recall = 0.59, F1-Score = 0.62
   - **Κλάση 1:**Precision = 0.62, Recall = 0.68, F1-Score = 0.65
2. **Overall:**
   - Accuracy = 0.63
   - Macro Avg = 0.63
   - Weighted Avg = 0.63
3. **ΣΥΜΠΕΡΑΣΜΑΤΑ**
   - Η ακρίβεια (accuracy) είναι 0.63.
   - Οι μετρήσεις precision, recall και F1-score είναι σχετικά ισορροπημένες μεταξύ των δύο κλάσεων.

Πυρήνας Ακτινικής Βάσης (RBF) (C=1, gamma='auto')

	precision	recall	f1-score	support
0	0.64	0.70	0.67	3696
1	0.67	0.61	0.64	3695
accuracy			0.65	7391
macro avg	0.65	0.65	0.65	7391
weighted avg	0.65	0.65	0.65	7391

Σχήμα 9.4: ΠΥΡΗΝΑΣ ΑΚΤΙΝΙΚΗΣ ΒΑΣΗΣ SVM

1. **Precision, Recall, F1-Score:**
    - **Κλάση 0:**Precision = 0.64, Recall = 0.70, F1-Score = 0.67
    - **Κλάση 1:**Precision = 0.67, Recall = 0.61, F1-Score = 0.64
2. **Overall:**
    - Accuracy = 0.65
    - Macro Avg = 0.65
    - Weighted Avg = 0.65
3. **ΣΥΜΠΕΡΑΣΜΑΤΑ**
    - Η απόδοση είναι ίδια με την "Out of the Box" περίπτωση.
    - Η ακρίβεια παραμένει στο 0.65 με ισορροπημένες μετρήσεις precision, recall και F1-score.

Σιγμοειδής Πυρήνας (C=0.01, kernel='sigmoid')

	precision	recall	f1-score	support
0	0.64	0.57	0.61	3696
1	0.62	0.68	0.65	3695
accuracy			0.63	7391
macro avg	0.63	0.63	0.63	7391
weighted avg	0.63	0.63	0.63	7391

Σχήμα 9.5: ΣΙΓΜΟΕΙΔΗΣ ΠΥΡΗΝΑΣ SVM

1. **Precision, Recall, F1-Score:**
   - **Κλάση 0:**Precision = 0.64, Recall = 0.57, F1-Score = 0.61
   - **Κλάση 1:**Precision = 0.62, Recall = 0.68, F1-Score = 0.65
2. **Overall:**
   - Accuracy = 0.63
   - Macro Avg = 0.63
   - Weighted Avg = 0.63
3. **ΣΥΜΠΕΡΑΣΜΑΤΑ**
   - Η ακρίβεια (accuracy) είναι 0.63.
   - Υπάρχει μια ισορροπία μεταξύ των μετρήσεων precision και recall για τις δύο κλάσεις.

9.2 MLP

MLP

Αποτελέσματα αρχικής εκπαίδευσης:

	precision	recall	f1-score	support
0	0.64	0.71	0.67	3696
1	0.67	0.60	0.64	3695
accuracy			0.66	7391
macro avg	0.66	0.66	0.66	7391
weighted avg	0.66	0.66	0.66	7391

Σχήμα 9.6: MLP ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΡΧΙΚΗΣ ΕΚΠΑΙΔΕΥΣΗΣ

1. **Precision, Recall, F1-Score:**
   - **Κλάση 0:**Precision = 0.64, Recall = 0.71, F1-Score = 0.67
   - **Κλάση 1:**Precision = 0.67, Recall = 0.60, F1-Score = 0.64
2. **Overall:**
   - Accuracy: 0.66
   - Macro Avg: 0.66
   - Weighted Avg: 0.66
3. **ΣΥΜΠΕΡΑΣΜΑΤΑ**
   - Η ακρίβεια (accuracy) είναι 0.66, με το μοντέλο να έχει ελαφρώς καλύτερη απόδοση στην κλάση 1 σε σχέση με την κλάση 0.
   - Το μοντέλο έχει ισορροπημένη απόδοση (macro και weighted averages είναι ίσα).

MLP

Αποτελέσματα μετά τη βελτιστοποίηση:

	precision	recall	f1-score	support
0	0.64	0.70	0.67	3696
1	0.67	0.60	0.63	3695
accuracy			0.65	7391
macro avg	0.65	0.65	0.65	7391
weighted avg	0.65	0.65	0.65	7391

Σχήμα 9.7: MLP ΑΠΟΤΕΛΕΣΜΑΤΑ ΜΕΤΑ ΤΗΝ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ

1. **Precision, Recall, F1-Score:**
   - **Κλάση 0:**Precision = 0.64, Recall = 0.70, F1-Score = 0.67
   - **Κλάση 1:**Precision = 0.67, Recall = 0.60, F1-Score = 0.63
2. **Overall:**
   - Accuracy: 0.65
   - Macro Avg: 0.65
   - Weighted Avg: 0.65
3. **ΣΥΜΠΕΡΑΣΜΑΤΑ**
   - Η ακρίβεια (accuracy) είναι 0.65, με το μοντέλο να έχει ελαφρώς καλύτερη απόδοση στην κλάση 1 σε σχέση με την κλάση 0.
   - Το μοντέλο έχει ισορροπημένη απόδοση (macro και weighted averages είναι ίσα).

9.3 CNN

```
Epoch 1/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 14s 74ms/step - accuracy: 0.2837 - loss: 1.9829 - val_accuracy: 0.9460 - val_loss: 0.1772
Epoch 2/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 13s 73ms/step - accuracy: 0.9380 - loss: 0.1860 - val_accuracy: 0.9719 - val_loss: 0.0831
Epoch 3/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 13s 76ms/step - accuracy: 0.9782 - loss: 0.0699 - val_accuracy: 0.9888 - val_loss: 0.0471
Epoch 4/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 13s 74ms/step - accuracy: 0.9829 - loss: 0.0540 - val_accuracy: 0.9899 - val_loss: 0.0427
Epoch 5/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 13s 73ms/step - accuracy: 0.9888 - loss: 0.0300 - val_accuracy: 0.9860 - val_loss: 0.0424
Epoch 6/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 13s 74ms/step - accuracy: 0.9953 - loss: 0.0175 - val_accuracy: 0.9727 - val_loss: 0.0837
Epoch 7/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 13s 74ms/step - accuracy: 0.9910 - loss: 0.0305 - val_accuracy: 0.9914 - val_loss: 0.0336
Epoch 8/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 13s 73ms/step - accuracy: 0.9927 - loss: 0.0249 - val_accuracy: 0.9903 - val_loss: 0.0379
Epoch 9/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 13s 72ms/step - accuracy: 0.9961 - loss: 0.0134 - val_accuracy: 0.9921 - val_loss: 0.0241
Epoch 10/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 12s 71ms/step - accuracy: 0.9947 - loss: 0.0124 - val_accuracy: 0.9924 - val_loss: 0.0291
Epoch 11/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 12s 71ms/step - accuracy: 0.9950 - loss: 0.0181 - val_accuracy: 0.9809 - val_loss: 0.0679
Epoch 12/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 12s 71ms/step - accuracy: 0.9934 - loss: 0.0241 - val_accuracy: 0.9896 - val_loss: 0.0382
Epoch 13/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 12s 71ms/step - accuracy: 0.9947 - loss: 0.0175 - val_accuracy: 0.9899 - val_loss: 0.0283
Epoch 14/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 13s 72ms/step - accuracy: 0.9950 - loss: 0.0140 - val_accuracy: 0.9921 - val_loss: 0.0307
Epoch 15/15
174/174 ━━━━━━━━━━━━━━━━━━━━ 12s 71ms/step - accuracy: 0.9956 - loss: 0.0129 - val_accuracy: 0.9888 - val_loss: 0.0410
```

Σχήμα 9.8: ΑΠΟΤΕΛΕΣΜΑΤΑ CNN

1. **Αξιολόγηση στο Σύνολο Επικύρωσης**
   - Η ακρίβεια στο σύνολο επικύρωσης ξεκίνησε από 0.9460 και έφτασε μέχρι το 0.9924.
   - Η απώλεια στο σύνολο επικύρωσης ξεκίνησε από 0.1772 και μειώθηκε μέχρι 0.0241.
2. **Σημάδια Υπερεκπαίδευσης (Overfitting)**
   - Στις τελευταίες εποχές, η ακρίβεια στο σύνολο εκπαίδευσης είναι πολύ υψηλή (σχεδόν 100%), ενώ η ακρίβεια στο σύνολο επικύρωσης παραμένει ελαφρώς χαμηλότερη.
   - Η απώλεια στο σύνολο επικύρωσης αυξάνεται ελαφρώς σε μερικές εποχές (π.χ., Epoch 6 και Epoch 11), υποδεικνύοντας πιθανά σημάδια υπερεκπαίδευσης.
3. **ΣΥΜΠΕΡΑΣΜΑΤΑ**
   - Η ακρίβεια **(accuracy)** στο σύνολο εκπαίδευσης έφτασε το 0.9956, δείχνοντας ότι το μοντέλο έχει εκπαιδευτεί καλά στα δεδομένα εκπαίδευσης.
   - Η απώλεια **(loss)** στο σύνολο επικύρωσης έφτασε το 0.0241, δείχνοντας ότι το μοντέλο έχει χαμηλή απόκλιση.

10. ΕΠΙΛΟΓΟΣ

Στην παρούσα πτυχιακή εργασία, εξετάσαμε τη χρήση μηχανικής μάθησης και τεχνητής νοημοσύνης για τη διαγνωστική και αναγνώριση στην αυτοκινητοβιομηχανία. Η έρευνα επικεντρώθηκε σε δύο βασικά μέρη: τη διάγνωση προβλημάτων κινητήρα και την αναγνώριση οδικών σημάτων.

Η εργασία ξεκίνησε με την εισαγωγή, όπου καθορίστηκαν οι στόχοι της πτυχιακής και τα εργαλεία που χρησιμοποιήθηκαν. Στη συνέχεια, ακολουθήθηκαν συγκεκριμένα βήματα προεπεξεργασίας δεδομένων. Για τον κινητήρα, η διαδικασία περιλάμβανε τη φόρτωση και προεπεξεργασία των δεδομένων, όπως καθαρισμό, oversampling, διαχωρισμό σε σύνολα εκπαίδευσης και δοκιμής, κανονικοποίηση και εξαγωγή χαρακτηριστικών. Για την αναγνώριση σημάτων, η προεπεξεργασία περιλάμβανε μετατροπή σε ασπρόμαυρη εικόνα, ισοστάθμιση ιστογράμματος και κανονικοποίηση.

Ακολούθως, αναλύθηκαν τα χαρακτηριστικά των δεδομένων κινητήρα και σημάτων και παρουσιάστηκαν με διάφορες οπτικοποιήσεις, όπως ιστογράμματα και διαγράμματα. Τα βασικά βήματα ανάλυσης και τα μοντέλα που χρησιμοποιήθηκαν περιλάμβαναν για τον κινητήρα MLP και SVM. Ακολουθήθηκαν βήματα όπως η φόρτωση δεδομένων, υπερδειγματοληψία, κανονικοποίηση, αρχική εκπαίδευση, αξιολόγηση και βελτιστοποίηση υπερπαραμέτρων. Για την αναγνώριση σημάτων, η διαδικασία περιλάμβανε φόρτωση και προετοιμασία δεδομένων, προεπεξεργασία εικόνων, διαχωρισμό δεδομένων, κατασκευή και εκπαίδευση μοντέλου, αποθήκευση και αξιολόγηση.

Η αρχική απόδοση του SVM χωρίς βελτιστοποίηση έδειξε μέτρια αποτελέσματα με ακρίβεια 0.65. Μετά από βελτιστοποίηση υπερπαραμέτρων, η απόδοση παρέμεινε σχεδόν ίδια, υποδεικνύοντας ότι οι αρχικές ρυθμίσεις ήταν ήδη κοντά στις βέλτιστες. Για τον κινητήρα MLP, η κανονικοποίηση και η υπερδειγματοληψία βοήθησαν στη βελτίωση της απόδοσης. Μετά από βελτιστοποίηση υπερπαραμέτρων, η απόδοση παρέμεινε σχεδόν ίδια, υποδεικνύοντας ότι οι αρχικές ρυθμίσεις ήταν ήδη κοντά στις βέλτιστες.. Στην αναγνώριση σημάτων, η χρήση τεχνικών προεπεξεργασίας εικόνας βελτίωσε την ακρίβεια και την απόδοση του μοντέλου.

Συμπερασματικά, η εργασία αυτή έθεσε τα θεμέλια για τη χρήση μηχανικής μάθησης και τεχνητής νοημοσύνης στην αυτοκινητοβιομηχανία, δείχνοντας τη δυναμική και τις προκλήσεις αυτών των τεχνολογιών. Ως επόμενα βήματα, προτείνεται η περαιτέρω βελτιστοποίηση υπερπαραμέτρων με διερεύνηση περισσότερων τιμών για C και gamma για rbf και sigmoid πυρήνες και δοκιμή διαφορετικών μεθόδων κανονικοποίησης και εξαγωγής χαρακτηριστικών. Επιπλέον, η συλλογή περισσότερων δεδομένων και η χρήση τεχνικών αύξησης δεδομένων μπορούν να ενισχύσουν τα σύνολα δεδομένων εικόνας. Τέλος, η εφαρμογή εναλλακτικών αλγορίθμων όπως Random Forest, Gradient Boosting και νευρωνικά δίκτυα μπορεί να οδηγήσει σε καλύτερα αποτελέσματα και εφαρμογές στην πράξη, ενσωματώνοντας επιπλέον χαρακτηριστικά από εξωτερικές πηγές ή μέσω τεχνικών deep learning.