Εργασία Εξαμήνου στη Πρακτική Μηχανική Μάθηση

Μαρία Μάστορα ΜΕ2026
Μεγάλα Δεδομένα και Αναλυτική
12/06/2021

Περιεχόμενα

Περίληψη

Η παρούσα εργασία αφορά το assortment planning/optimization. Δεδομένων δύο αρχείων με προϊόντα και χρήσιμες πληροφορίες για αυτά, μελετήθηκαν και απαντήθηκαν ιδιαίτερα ουσιώδη ζητήματα που μπορεί να αποτελούν χρήσιμα στον κλάδο του assortment analysis. Σε πρώτο στάδιο πραγματοποιήθηκε διερεύνηση των δεδομένων με σκοπό την προ-επεξεργασία τους με κατάλληλο τρόπο. Σε επόμενο στάδιο εφαρμόστηκαν μοντέλα μηχανικής μάθησης πρόβλεψης/Regression και κατηγοριοποίησης/Classification. Συγκεκριμένα χρησιμοποιήθηκαν τα μοντέλα Linear Regression, SVM, KNN και Neural Network για πρόβλεψη εσόδων, και τα μοντέλα Logistic Regression, KNN και SVM για κατηγοριοποίηση. Έπειτα, παρουσιάστηκαν κάποια αποτελέσματα και μέσω πινάκων και διαγραμμάτων, καθώς επίσης και σημαντικά συμπεράσματα σχετικά με την ανάλυση που πραγματοποιήθηκε.

Πρόβλημα

Η ανάλυση στηρίζεται και πραγματοποιείται με τον ίδιο τρόπο πάνω στα δύο σύνολα δεδομένων. Τα δύο αυτά σύνολα δεδομένων διαφέρουν στις διαστάσεις. Το πρώτο σύνολο περιέχει συνδυασμούς από συνολικά 17 προϊόντα όπου μπαίνουν σε 7άδες ως υποσύνολα και το δεύτερο σύνολο δεδομένων περιέχει συνδυασμούς από 20 συνολικά προϊόντα όπου μπαίνουν σε 20άδες.

Πιο αναλυτικά, δεδομένων $(m, \ell) \in \{(20, 10), (17, 7)\}$. Κάθε γραμμή αφορά ένα διαφορετικό υποσύνολο (assortment) ℓ προϊόντων από τα m, που δοκιμάστηκε στον πληθυσμό. Για κάθε προϊόν i στο υποσύνολο καταγράφονται (στην ίδια γραμμή, σειριακά, για όλα τα προϊόντα του υποσυνόλου):

- κωδικός προϊόντος $i = 1, 2, \ldots, m$,
- το ποσοστό αγοραστών που αγόρασαν το προϊόν i,
- το ποσοστό αγοραστών που αγόρασαν αποκλειστικά το προϊόν i,
- η μέση ποσοστιαία συνεισφορά του i στα μέσα έσοδα.

➢ Η τελευταία τιμή κάθε γραμμής σε κάθε αρχείο δεδομένων είναι ο μέσος όρος των εσόδων από το υποσύνολο προϊόντων που αντιστοιχεί στη γραμμή αυτή.

Όπως αναφέρεται και στη περιγραφή της εργασίας, τα ερωτήματα χωρίζονται σε δύο σκέλη.

(Α) Για δεδομένο προϊόν, ως μέλος ενός υποσυνόλου ℓ προϊόντων από τα m:

1. πρόβλεψη αναμενόμενων εσόδων από το προϊόν αυτό,
2. αν τα αναμενόμενα έσοδα υπερβαίνουν ή όχι τη μέση τιμή αναμενόμενων εσόδων από το προϊόν αυτό (υπολογιζόμενη πάνω από τα υποσύνολα προϊόντων ενός training set, που περιέχουν το προϊόν αυτό).

(Β) Για συγκεκριμένο υποσύνολο ℓ προϊόντων από τα m:

1. πρόβλεψη αναμενόμενων εσόδων,
2. αν τα αναμενόμενα έσοδα υπερβαίνουν ή όχι τη μέση τιμή αναμενόμενων εσόδων (υπολογιζόμενη πάνω από δεδομένα υποσύνολα προϊόντων σε ένα training set).

Μεθοδολογική Προσέγγιση

Πρώτο Dataset 17 συνολικών προϊόντων σε υποσύνολα των επτάδων

A1) Προ-επεξεργασία και εφαρμογή Regression μοντέλων

Το σύνολο δεδομένων, με κατάλληλες ονομασίες στις στήλες, περιέχει τις ακόλουθες πληροφορίες για τα προϊόντα όπως απεικονίζονται στη παρακάτω εικόνα [Εικόνα 1].

	p1	p1_buyers_perc	p1_only_buyers_perc	p1_percentage_contribution	...	p7	p7_buyers_perc	p7_only_buyers_perc	p7_percentage_contribution	average_income
0	0	84.00	0.0	20.43	...	14	52.00	0.0	6.81	425.8133
1	2	45.33	0.0	4.38	...	13	100.00	12.0	42.07	409.8400
2	3	56.00	0.0	10.83	...	14	42.67	0.0	6.00	365.1733
3	2	38.67	0.0	3.85	...	12	21.33	0.0	1.56	380.3200
4	3	81.33	0.0	24.34	...	16	64.00	0.0	16.56	298.6933
...	...	...	...	...	...	...	...	...	...	...
7837	1	66.67	0.0	16.01	...	13	86.67	0.0	26.24	466.0000
7838	0	68.00	0.0	13.84	...	14	38.67	0.0	4.73	425.0133
7839	3	45.33	0.0	6.74	...	15	30.67	0.0	4.04	468.6667
7840	0	89.33	0.0	22.61	...	10	73.33	0.0	15.66	432.6933
7841	0	80.00	0.0	21.06	...	15	38.67	0.0	5.92	390.8000

Εικόνα 1. Αρχικό DataFrame

Έχοντας το DataFrame αυτό, παρατηρήθηκε ότι τα 17 συνολικά προϊόντα εμφανίζονται σε 7άδες στα υποσύνολα-γραμμές. Στο σημείο αυτό αποφασίστηκε να χρησιμοποιηθούν οι στήλες των προϊόντων, καθώς επίσης και να υπολογισθούν τα έσοδα καθενός από τα προϊόντα σε καθένα από τα υποσύνολα.

Σε αυτό το DataFrame, στη συνέχεια προστέθηκε η στήλη *SetID*, έτσι ώστε να καταχωρείται σε αυτή τη στήλη η πληροφορία του μοναδικού κωδικού του κάθε υποσυνόλου προϊόντων.

Συνεπώς, για τους σκοπούς του πρώτου ερωτήματος, υπολογίσθηκαν τα έσοδα καθενός από τα προϊόντα που βρίσκεται σε καθένα από τα υποσύνολα, δεδομένου του μέσου όρου εσόδων κάθε υποσυνόλου και του ποσοστού συνεισφοράς του προϊόντος στο υποσύνολο, όπου δίνονται ως πληροφορία στις αντίστοιχες στήλες του DataFrame. Οι επιπλέον στήλες με τα έσοδα καθενός προϊόντος, δημιουργήθηκαν μέσω του ακόλουθου τύπου:

```
prod_with_incomes['p7_income']=
products_subsets['p7_percentage_contribution']*products_subsets['average_income'] / 100
```

Έτσι, το DataFrame διαθέτει την μορφή όπως παρουσιάζεται στην παρακάτω εικόνα [Εικόνα 2].

	SetID	p1_income	p2_income	p3_income	p4_income	p5_income	p6_income	p7_income	p1	p2	p3	p4	p5	p6	p7
0	0	86.993657	14.137002	47.350439	2.725205	183.057138	62.551974	28.997886	0	2	3	4	7	10	14
1	1	17.950992	1.885264	46.721760	8.606640	58.361216	103.853456	172.419688	2	4	6	8	9	11	13
2	2	39.548268	2.921386	51.708539	77.854948	159.690284	6.719189	21.910398	3	8	9	10	11	12	14
3	3	14.642320	1.407184	41.036528	161.445840	57.998800	87.701792	5.932992	2	4	6	7	9	10	12
4	4	72.701949	4.599877	1.672682	10.693220	141.132584	14.456756	49.463610	3	4	5	8	11	12	16
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
7837	7837	74.606600	8.900600	26.515400	172.000600	2.982400	46.273800	122.278400	1	2	6	7	8	10	13
7838	7838	58.821841	107.528365	9.350293	41.948813	165.755187	4.547642	20.103129	0	1	2	3	7	12	14
7839	7839	31.588136	44.898270	177.390346	65.707071	2.812000	121.150342	18.934135	3	6	7	10	12	13	15
7840	7840	97.831955	167.755192	12.375028	34.052963	2.250005	44.870295	67.759771	0	1	2	3	5	6	10
7841	7841	82.302480	180.119720	8.832080	34.312240	52.875240	3.986160	23.135360	0	1	2	6	10	12	15

Εικόνα 2. DataFrame με τα προϊόντα και τα έσοδα που έχουν σε καθένα από τα υποσύνολα

Στη συνέχεια, χρησιμοποιήθηκε η συνάρτηση *pandas.melt* με τη βοήθεια της οποίας πραγματοποιήθηκε αποσύνθεση του DataFrame σε μεγαλύτερο αριθμό γραμμών. Η διαδικασία αυτή υλοποιήθηκε λόγω του ότι χρειάστηκε η πληροφορία των δεδομένων *SetID–prod_id–prod_income* ανά γραμμή. Με πιο απλά λόγια, με τον τρόπο της αποσύνθεσης, λήφθηκε η πληροφορία για κάθε υποσύνολο προϊόντων (*p1,..,p7*), για κάθε προϊόν (*prod_id*), το εισόδημα που λαμβάνεται (*prod_income*) στο συγκεκριμένο υποσύνολο, σε νέα στήλη.

Πιο αναλυτικά, μετά το DataFrame της εικόνας 2, ακολούθησαν τρία βήματα.

- Εφαρμόστηκε η συνάρτηση *melt* στο DataFrame αυτό, έτσι ώστε να καταχωρούνται οι τιμές των εσόδων από κάθε προϊόν, σε νέα στήλη με ονομασία *value*. Το αποτέλεσμα εκχωρήθηκε σε DataFrame που ονομάστηκε *melted_df1*.
- Εφαρμόστηκε η συνάρτηση *melt* στο DataFrame αυτό, με διαφορετικές παραμέτρους, έτσι ώστε να καταχωρούνται σε νέα στήλη τα προϊόντα για τα οποία υπολογίζονται τα έσοδα σε καθένα υποσύνολο. Η στήλη αυτή δημιουργήθηκε με ονομασία *prod_id*. Το αποτέλεσμα εκχωρήθηκε σε DataFrame που ονομάστηκε *melted_df2*.
- Εφαρμόστηκε η συνάρτηση *concat* στα δύο υπολογιζόμενα DataFrames *melted_df1*, *melted_df2*, με σκοπό την συνένωσή τους σε ένα νέο.

Με τα παραπάνω βήματα και με φιλτράρισμα των στηλών όπου χρειάστηκαν, προέκυψε το DataFrame *products_subsets_model* όπου και χρησιμοποιήθηκε στην εφαρμογή των μοντέλων. Στην παρακάτω εικόνα [Εικόνα 3] παρουσιάζεται η μορφή του DataFrame αυτού.

	p1	p2	p3	p4	p5	p6	p7	prod_income	prod_id
0	0	2	3	4	7	10	14	86.993657	0
7842	0	2	3	4	7	10	14	14.137002	2
15684	0	2	3	4	7	10	14	47.350439	3
23526	0	2	3	4	7	10	14	2.725205	4
31368	0	2	3	4	7	10	14	183.057138	7
...	...	...	...	...	...	...	...	...	...
23525	0	1	2	6	10	12	15	8.832080	2
31367	0	1	2	6	10	12	15	34.312240	6
39209	0	1	2	6	10	12	15	52.875240	10
47051	0	1	2	6	10	12	15	3.986160	12
54893	0	1	2	6	10	12	15	23.135360	15

54894 rows × 9 columns

Εικόνα 3. products_subsets_model DataFrame

Παρατηρήθηκε, ότι πράγματι λαμβάνονται όλοι οι συνδυασμοί προϊόντων στα υποσύνολα και πλέον ο αριθμός των γραμμών υφίσταται 7842×7, δηλαδή ίσος με 54894. Συνεπώς, πλέον το DataFrame περιέχει τα προϊόντα από κάθε υποσύνολο, το προϊόν για το οποίο υπολογίσθηκαν τα έσοδα πάνω σε αυτό το υποσύνολο, και το ποσό των εσόδων από το προϊόν αυτό.

Σε επόμενο στάδιο, δοκιμάστηκε η εφαρμογή μοντέλων μηχανικής μάθησης πάνω στο DataFrame της εικόνας 3, και παρατηρήθηκε ότι δεν καταγράφηκαν ικανοποιητικά υψηλές τιμές πρόβλεψης. Συνεπώς, αποφασίστηκε χρήση της συνάρτησης *pandas.get_dummies*, για μετατροπή των κατηγορικών τιμών *p1,p2,p3,p4,p5,p6,p7,prod_id* σε μεταβλητές κωδικοποιημένες-encoding σε συνδυασμούς 0-1. Η μορφή του DataFrame παρουσιάζεται στην παρακάτω εικόνα [Εικόνα 4] με δείγμα 2 γραμμών.

	prod_income	prod_id_0	prod_id_1	prod_id_2	prod_id_3	prod_id_4	prod_id_5	prod_id_6	prod_id_7	prod_id_8	...
0	86.993657	1	0	0	0	0	0	0	0	0	...
7842	14.137002	0	0	1	0	0	0	0	0	0	...

Εικόνα 4. Encoded DataFrame για την εφαρμογή των μοντέλων με τα προϊόντα και τα έσοδα τους

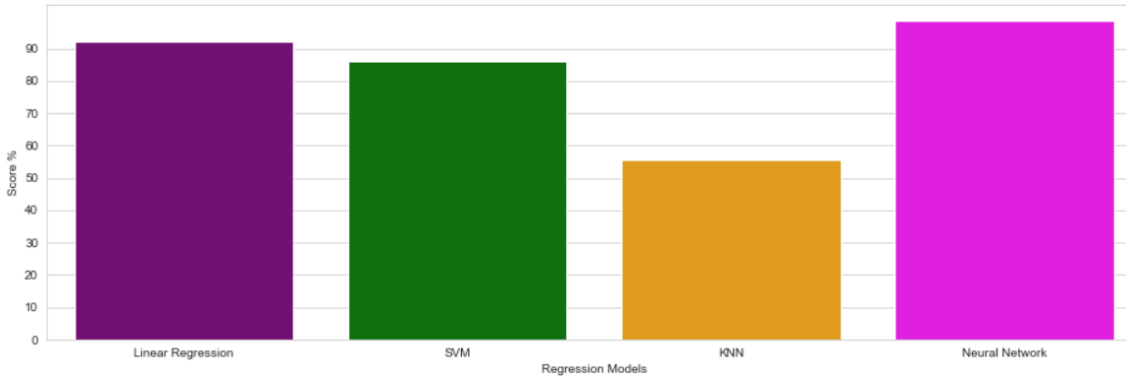
Αλλάζοντας, έτσι τις συγκεκριμένες στήλες κατά αυτόν τον τρόπο και αφήνοντας απαράλλακτη τη στήλη *prod_income*, προέκυψε το DataFrame, πάνω στο οποίο εφαρμόστηκαν τα μοντέλα μηχανικής μάθησης.

Τέθηκαν οι στήλες των προϊόντων *p1,p2,p3,p4,p5,p6,p7,prod_id* (ανεξάρτητες μεταβλητές) ως μεταβλητές εισόδων X, και η στήλη των εσόδων *prod_income* (εξαρτημένη μεταβλητή) ως μεταβλητή εξόδου y. Εφαρμόστηκε η τεχνική Split Data, μέσω της συνάρτησης *train_test_split()* της βιβλιοθήκης *sklearn* όπου τα δεδομένα χωρίστηκαν σε 80% για training και 20% για test.

Εφαρμόστηκαν τα ακόλουθα Regression μοντέλα μηχανικής μάθησης: Linear Regression, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Neural Network (MLP Regressor). Στον παρακάτω πίνακα [Πίνακας 1] απεικονίζονται οι τιμές Score (r2 score) και MSE (Mean Square Error) που κατέγραψαν τα μοντέλα αυτά.

	Linear Regression	**SVM**	**KNN**	**Neural Network**
Score	0.92	0.86	0.55	0.98
MSE	231.66	411.71	1349.96	41.28

Πίνακας 1. Score και MSE των Regression Μοντέλων



Εικόνα 5. Ραβδόγραμμα των τιμών Score για τα Regression μοντέλα

Α2) Προ-επεξεργασία και εφαρμογή Classification μοντέλων

Στο ερώτημα αυτό ζητήθηκε εάν τα αναμενόμενα έσοδα από ένα δεδομένο προϊόν υπερβαίνουν ή όχι τη μέση τιμή αναμενόμενων εσόδων από το προϊόν αυτό. Το ερώτημα αυτό δοκιμάστηκε με δύο τρόπους, οι οποίοι αναλύονται στη συνέχεια.

Α Τρόπος (χωρίς encode για τις κατηγορικές τιμές των προϊόντων)

Έχοντας το DataFrame της εικόνας 3, με στήλες *prod_id, p1, p2, p3, p4, p5, p6, p7, prod_income*, σε πρώτο στάδιο χωρίστηκαν οι μεταβλητές εισόδου X (*prod_id, p1, p2, p3, p4, p5, p6, p7*) με τη μεταβλητή εξόδου y (*prod_income*).

Έπειτα, χωρίστηκαν τα δεδομένα σε train set και test set, με χρήση της *train_test_split()* της βιβλιοθήκης *sklearn* (80% και 20% αντίστοιχα). Εν συνεχεία, έγινε το φιλτράρισμα του train set ως νέο DataFrame με ονομασία *training_set*, και έπειτα υπολογίσθηκε η μέση τιμή εσόδων στα υποσύνολα για καθένα από τα προϊόντα, πάνω στο DataFrame αυτό. Η διαδικασία αυτή επιτεύχθηκε με τη χρήση των συναρτήσεων *groupby* και *transform*, για τον υπολογισμό του μέσου όρου των εσόδων στα υποσύνολα για κάθε προϊόν.

Απλούστερα, το νέο DataFrame *training_set* περιέχει τα προϊόντα καθενός συνόλου (*p1,..,p7*), το προϊόν για το οποίο υπολογίσθηκαν τα έσοδα (*prod_id*), το ποσό των εσόδων από το προϊόν αυτό στο συγκεκριμένο υποσύνολο (*prod_income*), καθώς και τον μέσο όρο των εσόδων που έχει το συγκεκριμένο προϊόν στα υποσύνολα του training set (*mean_income_per_prod*). Παραδειγματικά, στην παρακάτω εικόνα [Εικόνα 6] απεικονίζεται το DataFrame με ονομασία training_set με τις πρώτες δύο γραμμές.

	prod_id	p1	p2	p3	p4	p5	p6	p7	prod_income	mean_income_per_prod
8680	4	1	4	6	10	11	12	15	1.562661	2.994327
27026	10	3	4	7	10	13	14	16	72.955326	72.824550

Εικόνα 6. training_set DataFrame

Στη συνέχεια, δημιουργήθηκε νέο DataFrame με το test set με ονομασία test_set περιέχοντας τα προϊόντα ενός υποσυνόλου-για κάθε γραμμή (*p1,..,p7*), το προϊόν για το οποίο υπολογίσθηκαν τα έσοδα (*prod_id*) και το ποσό των εσόδων από το προϊόν αυτό στο συγκεκριμένο υποσύνολο (*prod_income*).

Έπειτα, δημιουργήθηκε νέο DataFrame με στήλες τα prod_id και το μέσο όρο των εσόδων που έχει καθένα από αυτά τα 17, όπου υπολογίστηκαν πάνω στο training set όπως αναφέρθηκε. Σε επόμενο στάδιο, έγινε ένωση (μέσω της συνάρτησης *merge*) αυτού του DataFrame με το DataFrame test_set, έτσι ώστε να υπάρχει η πληροφορία των προϊόντων (*p1,..,p7*), των εσοδών (*prod_income*), και του μέσου όρου των εσόδων του prod_id όπου υπολογίσθηκε πάνω στο training set (*mean_income_per_prod*).

Στη συνέχεια, τέθηκαν ως label νέες binary τιμές με τη σύγκριση των στηλών *prod_income* και *mean_income_per_prod* και στο training και στο test set, για την μετέπειτα πρόβλεψη των μοντέλων. Για τον λόγο του ότι η τιμή της πρόβλεψης είναι binary (0-1), πραγματοποιήθηκε μετατροπή στη μεταβλητή εξόδου σε binary ανάλογα με το αν για κάθε υποσύνολο προϊόντων τα έσοδα κάθε προϊόντος υπερβαίνουν (ή όχι) τη μέση τιμή που υπολογίσθηκε. Έτσι, τέθηκαν νέα Labels για τιμές εξόδου με binary τιμές και για το train set και για το test set, με τη χρήση της συνάρτησης *astype()*.

Στη συνέχεια, εφαρμόστηκαν τα εξής μοντέλα Classification: Logistic Regression, KNN, SVM. Οι τιμές accuracy scores που καταγράφηκαν απεικονίζονται στον παρακάτω πίνακα [Πίνακας 2]:

Logistic Regression (Accuracy)	**KNN (Accuracy)**	**SVM (Accuracy)**
57.16 %	74.92 %	57.20 %

Πίνακας 2. Accuracy scores για τα Classification μοντέλα χωρίς Encode

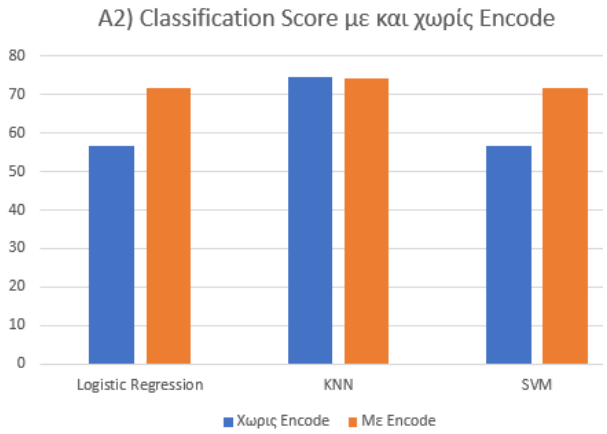
Β Τρόπος (με encode για τις κατηγορικές τιμές των προϊόντων)

Το ερώτημα αυτό επιτεύχθηκε με τον ίδιο τρόπο όπως και στον Α τρόπο, με τη διαφορά ότι έγινε χρήση της τεχνικής κωδικοποίησης (encoding) με τη συνάρτηση *get_dummies*, πάνω στις στήλες των προϊόντων.

Έτσι έπειτα από την διαδικασία επεξεργασίας, την αλλαγή των Labels σε binary τιμές, και την εφαρμογή των μοντέλων Classification Logistic Regression, KNN, SVM, προκύπτουν οι τιμές accuracy όπως παρουσιάζονται στο παρακάτω πίνακα [Πίνακας 3].

Logistic Regression (Accuracy)	**KNN (Accuracy)**	**SVM (Accuracy)**
71.95 %	74.66 %	71.91 %

Πίνακας 3. Accuracy scores για τα Classification μοντέλα με Encode



Εικόνα 7. Ραβδόγραμμα με σύγκριση των Accuracy με και χωρίς encode.

Β1) Προ-επεξεργασία και εφαρμογή Regression μοντέλων

Στο ερώτημα αυτό για συγκεκριμένο υποσύνολο ℓ προϊόντων από τα m, ζητήθηκε να γίνει πρόβλεψη των αναμενόμενων εσόδων.

Για την υλοποίηση της διαδικασίας σε αυτό το ερώτημα, χρησιμοποιήθηκαν οι στήλες των προϊόντων (*p1,..,p7*) και η τελευταία τιμή κάθε γραμμής στο αρχείο που δόθηκε (*average_income*), όπου είναι ο μέσος όρος των εσόδων από το υποσύνολο προϊόντων που αντιστοιχεί στη γραμμή αυτή. Στην παρακάτω εικόνα [Εικόνα 8] παρουσιάζεται το DataFrame με τα απαιτούμενα χαρακτηριστικά.

	p1	p2	p3	p4	p5	p6	p7	average_income
0	0	2	3	4	7	10	14	425.8133
1	2	4	6	8	9	11	13	409.8400
2	3	8	9	10	11	12	14	365.1733
3	2	4	6	7	9	10	12	380.3200
4	3	4	5	8	11	12	16	298.6933
...	...	...	...	...	...	...	...	...
7837	1	2	6	7	8	10	13	466.0000
7838	0	1	2	3	7	12	14	425.0133
7839	3	6	7	10	12	13	15	468.6667
7840	0	1	2	3	5	6	10	432.6933
7841	0	1	2	6	10	12	15	390.8000

Εικόνα 8. DataFrame με τις απαραίτητες στήλες για την εφαρμογή των μοντέλων

Σε πρώτο στάδιο εφαρμόστηκε κωδικοποίηση (encoding) στις στήλες των προϊόντων (*p1,..,p7*) με τη μέθοδο *get_dummies*. Το DataFrame πριν την εφαρμογή των μοντέλων έχει την μορφή όπως φαίνεται στην παρακάτω εικόνα [Εικόνα 9].

	average_income	p1_0	p1_1	p1_2	p1_3	p1_4	p1_5	p1_6	p1_7	p1_8	...
0	425.8133	1	0	0	0	0	0	0	0	0	...
1	409.8400	0	0	1	0	0	0	0	0	0	...
2	365.1733	0	0	0	1	0	0	0	0	0	...
3	380.3200	0	0	1	0	0	0	0	0	0	...
4	298.6933	0	0	0	1	0	0	0	0	0	...
...	...	...	...	...	...	...	...	...	...	...	...
7837	466.0000	0	1	0	0	0	0	0	0	0	...
7838	425.0133	1	0	0	0	0	0	0	0	0	...
7839	468.6667	0	0	0	1	0	0	0	0	0	...
7840	432.6933	1	0	0	0	0	0	0	0	0	...
7841	390.8000	1	0	0	0	0	0	0	0	0	...

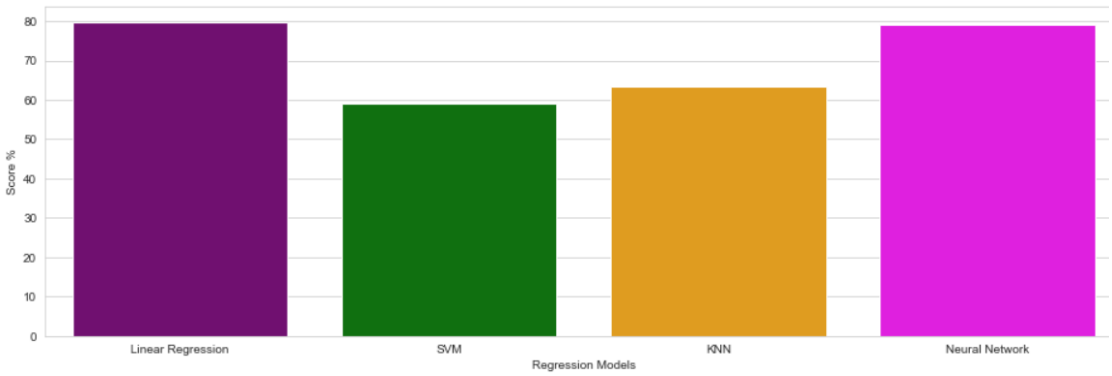
Εικόνα 9. DataFrame έπειτα από τη χρήση encoding στις κατάλληλες στήλες

Έχοντας τα encoded προϊόντα *p1,..,p7,* ως τιμές εσόδου X (ανεξάρτητες μεταβλητές) και τα μέσα έσοδα *average_income* (εξαρτημένη μεταβλητή) για καθένα από τα υποσύνολα ως τιμή εξόδου y, εφαρμόστηκε η μέθοδος Split Data μέσω της συνάρτησης *train_test_split()* και με 80% για training και 20% για test.

Εφαρμόστηκαν τα ακόλουθα μοντέλα Regression: Linear Regression, SVM, KNN, Neural Network. Οι τιμές των Score (r2 score) και MSE (Mean Square Error) καταγράφονται στον παρακάτω πίνακα [Πίνακας 4].

	Linear Regression	**SVM**	**KNN**	**Neural Network**
Score	0.79	0.59	0.63	0.79
MSE	433.92	875.68	766.66	447.20

Πίνακας 4. Score και MSE των Regression μοντέλων



Εικόνα 10. Ραβδόγραμμα των τιμών Score για τα Regression μοντέλα

B2) Προ-επεξεργασία και εφαρμογή Classification μοντέλων

Στο ερώτημα αυτό ζητήθηκε αν τα αναμενόμενα έσοδα υπερβαίνουν ή όχι τη μέση τιμή αναμενόμενων εσόδων. Το ερώτημα αυτό δοκιμάστηκε με δύο τρόπους, οι οποίοι και αναλύονται στη συνέχεια.

Α Τρόπος (χωρίς encode για τις κατηγορικές τιμές των προϊόντων)

Αρχικά, έχοντας το DataFrame με στήλες *p1, p2, p3, p4, p5, p6, p7, average_income*, και θέτοντας τις στήλες των προϊόντων (*p1,..,p7*) ως μεταβλητές εισόδου X – ανεξάρτητες μεταβλητές και τη στήλη των εσόδων (*average_income*) ως μεταβλητή εξόδου y - εξαρτημένη μεταβλητή, εφαρμόστηκε η μέθοδος Split Data με χρήση της συνάρτησης *train_test_split()* της *sklearn* με 80% training και 20% test set. Για τον λόγο του ότι ζητήθηκε αν τα αναμενόμενα έσοδα υπερβαίνουν (ή όχι) τη μέση τιμή αναμενόμενων εσόδων, αποφασίστηκε να τεθεί Label με binary τιμές.

Σε πρώτο στάδιο χωρίστηκαν σε διαφορετικά DataFrames το training και το test set και έπειτα υπολογίσθηκε η μέση τιμή των αναμενόμενων εσόδων πάνω στο training set. Συνεπώς, η μέση τιμή των εσόδων που υπολογίσθηκε στο training set εκχωρήθηκε στη μεταβλητή *mean_train_income* και στη συνέχεια με κατάλληλη συνθήκη, μετατράπηκαν οι τιμές εξόδου σε binary. Η μετατροπή των Labels σε binary τιμές (1-0) επιτεύχθηκε με τη χρήση της συνάρτησης *astype()*.

Έπειτα, εφαρμόστηκαν τα ακόλουθα μοντέλα Classification: Logistic Regression, KNN, SVM. Οι τιμές accuracy όπου καταγράφηκαν παρουσιάζονται στον παρακάτω πίνακα [Πίνακας 5].

Logistic Regression (Accuracy)	**KNN (Accuracy)**	**SVM (Accuracy)**
57.87 %	71.63 %	58.18 %

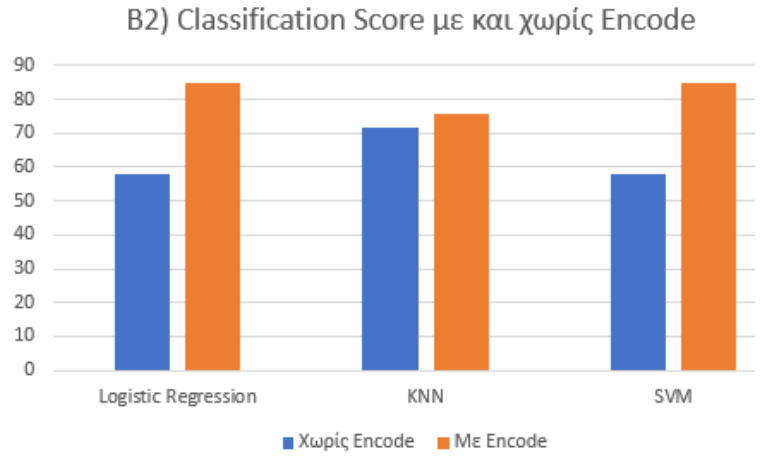
Πίνακας 5. Accuracy scores για τα Classification μοντέλα χωρίς Encode

Β Τρόπος (με encode για τις κατηγορικές τιμές των προϊόντων)

Το ερώτημα αυτό επιτεύχθηκε με τον ίδιο τρόπο όπως εξηγήθηκε προηγουμένως (με τον Α τρόπο) με την αλλαγή μέσω χρήσης κωδικοποίησης (encoding) στις στήλες των προϊόντων (*p1,..,p7*). Η κωδικοποίηση στις συγκεκριμένες στήλες επιτεύχθηκε μέσω της συνάρτησης *get_dummies*. Συνεπώς, έπειτα από την αλλαγή των Labels σε binary τιμές, και την εφαρμογή των μοντέλων Classification, προκύπτουν οι ακρίβειες όπως φαίνεται στο παρακάτω πίνακα [Πίνακας 6].

Logistic Regression (Accuracy)	**KNN (Accuracy)**	**SVM (Accuracy)**
85.08 %	75.78 %	84.83 %

Πίνακας 6. Accuracy scores για τα Classification μοντέλα με Encode



Εικόνα 11. Ραβδόγραμμα με σύγκριση των Accuracy με και χωρίς encode

Δεύτερο Dataset 20 συνολικών προϊόντων σε υποσύνολα των δεκάδων

Η ανάλυση για το δεύτερο σύνολο δεδομένων όπου περιέχει 20 συνολικά προϊόντα σε υποσύνολα των 10άδων έγινε με όμοιο τρόπο όπως στο πρώτο σύνολο δεδομένων με τις μικρότερες διαστάσεις. Εφαρμόστηκε η κατάλληλη προ-επεξεργασία για τις απαιτήσεις των ερωτημάτων, όπως επεξηγήθηκε παραπάνω και στο μικρότερο σύνολο δεδομένων.

A1) Προ-επεξεργασία και εφαρμογή Regression μοντέλων

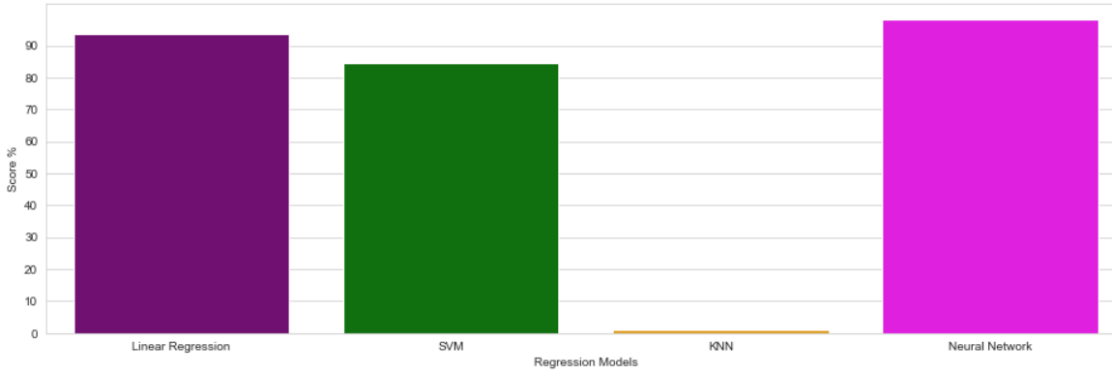
Για δεδομένο προϊόν, ως μέλος ενός υποσυνόλου ℓ προϊόντων από τα m, και την πρόβλεψη αναμενόμενων εσόδων από το προϊόν αυτό, εφαρμόστηκαν τέσσερα Regression μοντέλα μηχανικής μάθησης.

Όπως επεξηγήθηκε στο πρώτο σύνολο δεδομένων, δημιουργήθηκε DataFrame με τα υποσύνολα των 10 προϊόντων (*p1,..,p10*) μαζί με τα έσοδα που έχει κάθε προϊόν για κάθε σύνολο προϊόντων στο οποίο εμφανίζεται (*prod_income*), καθώς επίσης και το αναφερόμενο προϊόν για το οποίο υπολογίσθηκαν τα έσοδα (*prod_id*). Έπειτα από την εφαρμογή της μεθόδου *get_dummies* πάνω στα προϊόντα, εφαρμόστηκαν τα μοντέλα Linear Regression, SVM, KNN, Neural Network.

Παρατηρήθηκε, ότι πράγματι λαμβάνονται όλοι οι συνδυασμοί προϊόντων στα υποσύνολα και πλέον ο αριθμός των γραμμών υφίσταται 18940×10, δηλαδή ίσος με 189400. Στον παρακάτω πίνακα [Πίνακας 7] παρουσιάζονται οι τιμές Score (r2 score) και MSE (Mean Square Error) για αυτά.

	Linear Regression	**SVM**	**KNN**	**Neural Network**
Score	0.93	0.84	0.01	0.98
MSE	1987.42	4781.27	30879.87	549.53

Πίνακας 7. Score και MSE των Regression μοντέλων



Εικόνα 12. Ραβδόγραμμα των τιμών Score για τα Regression μοντέλα

Α2) Προ-επεξεργασία και εφαρμογή Classification μοντέλων

Με όμοιο τρόπο, όπως και στο πρώτο σύνολο δεδομένων, πραγματοποιήθηκε ανάλυση για το πρόβλημα κατηγοριοποίησης και συγκεκριμένα, αν από δεδομένο προϊόν, τα αναμενόμενα έσοδα υπερβαίνουν ή όχι τη μέση τιμή αναμενόμενων εσόδων από το προϊόν αυτό. Δοκιμάστηκαν έτσι, δύο τρόποι για την επίλυση του ερωτήματος, και εφαρμόστηκαν τρία Classification μοντέλα.

Α Τρόπος (χωρίς encode για τις κατηγορικές τιμές των προϊόντων)

Κατά τον τρόπο αυτό, τα μοντέλα μηχανικής μάθησης εκπαιδεύτηκαν με κατηγορικές μεταβλητές εισόδου, χωρίς τη τεχνική κωδικοποίησης (encoding) στις στήλες των προϊόντων.

Με τη βοήθεια των συναρτήσεων *groupby* και *transform*, υπολογίσθηκε η μέση τιμή των αναμενόμενων εσόδων από το κάθε προϊόν που εμφανίζεται στα υποσύνολα προϊόντων, πάνω στο train set. Έτσι, δημιουργήθηκε νέο DataFrame που περιέχει τα προϊόντα καθενός συνόλου (*p1,..,p10*), το προϊόν για το οποίο υπολογίσθηκαν τα έσοδα (*prod_id*), επιπλέον το ποσό των εσόδων από το προϊόν αυτό στο συγκεκριμένο υποσύνολο (*prod_income*), καθώς και τον μέσο όρο των εσόδων που έχει το συγκεκριμένο προϊόν στα υποσύνολα του training set (*mean_income_per_prod*).

Έπειτα, δημιουργήθηκε DataFrame με 20 γραμμές όσες και τα *prod_id* και τον μέσο όρο των εσόδων στα υποσύνολα, όπου υπολογίσθηκε πάνω στο training set. Στη συνέχεια έγινε ένωση αυτού του DataFrame με το *test_set* με βάση το *prod_id.* Η μορφή του νέου DataFrame που δημιουργήθηκε απεικονίζεται στη παρακάτω εικόνα [Εικόνα 13].

	prod_id	mean_income_per_prod	p1	p2	p3	p4	p5	p6	p7	prod_income
0	0	75.928124	0	3	8	10	11	12	14	57.190235
1	0	75.928124	0	3	7	8	9	10	11	34.350912
2	0	75.928124	0	3	4	8	10	13	14	53.860230
3	0	75.928124	0	1	2	3	9	10	14	105.714568
4	0	75.928124	0	1	3	10	13	15	16	45.134045
...	...	...	...	...	...	...	...	...	...	...
10974	16	37.377932	2	4	7	11	12	14	16	56.431272
10975	16	37.377932	3	4	6	7	9	15	16	31.781459
10976	16	37.377932	0	1	2	5	8	13	16	33.952843
10977	16	37.377932	1	4	6	9	10	13	16	14.656694
10978	16	37.377932	1	5	8	9	12	14	16	61.954120

Εικόνα 13. test set με το μέσο όρο των εσόδων που υπολογίσθηκε πάνω στο training set

Για τον λόγο του ότι οι τιμές εξόδου κλήθηκαν να είναι binary (υπερβαίνουν τη μέση τιμή-1, δεν υπερβαίνουν τη μέση τιμή-0), έγινε αλλαγή της Label τιμής για το σύνολο δεδομένων σε 0-1 με τη χρήση της συνάρτησης *astype()*. Στη συνέχεια, εφαρμόστηκαν τα τρία Classification μοντέλα μηχανικής μάθησης: Logistic Regression, KNN και SVM. Στο παρακάτω πίνακα [Πίνακας 8] παρουσιάζονται οι τιμές accuracy για καθένα από αυτά.

Logistic Regression (Accuracy)	KNN (Accuracy)	SVM (Accuracy)
58.95 %	69.67 %	58.57 %

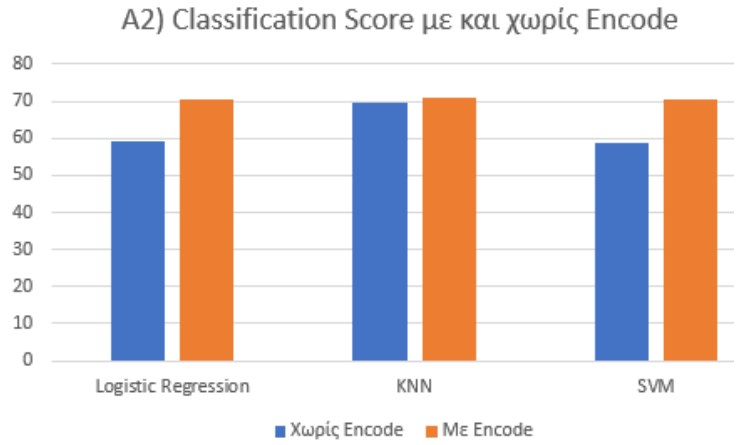
Πίνακας 8. Accuracy scores για τα Classification μοντέλα χωρίς Encode

Β Τρόπος (με encode για τις κατηγορικές τιμές των προϊόντων)

Η εφαρμογή των μοντέλων έγινε με τον ίδιο τρόπο όπως στον Α τρόπο, με την διαφορά ότι εφαρμόστηκε η τεχνική κωδικοποίησης (encoding), με χρήση της συνάρτησης *get_dummies* πάνω στις στήλες των προϊόντων. Τα αποτελέσματα των Classification μοντέλων απεικονίζονται στον παρακάτω πίνακα [Πίνακας 9].

Logistic Regression (Accuracy)	KNN (Accuracy)	SVM (Accuracy)
70.34 %	71.13 %	70.41 %

Πίνακας 9. Accuracy scores για τα Classification μοντέλα με Encode



Εικόνα 14. Ραβδόγραμμα με σύγκριση των Accuracy με και χωρίς encode

Β1) Προ-επεξεργασία και εφαρμογή Regression μοντέλων

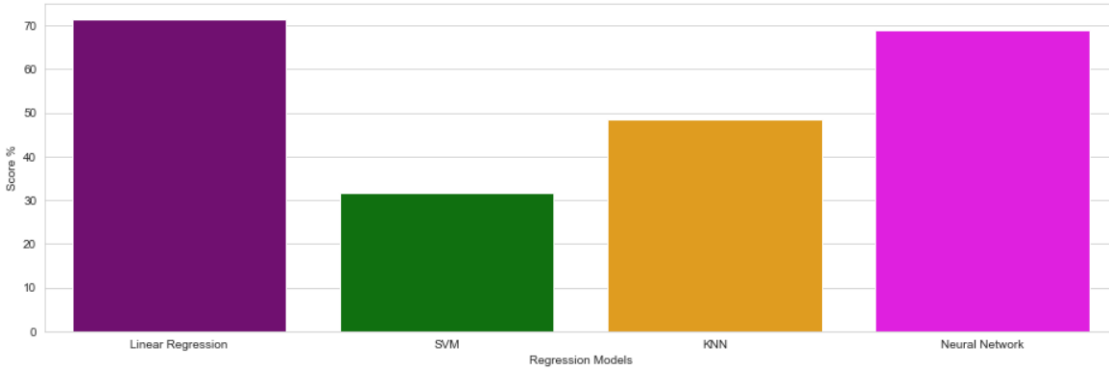
Για τις ανάγκες του ερωτήματος αυτού χρησιμοποιήθηκαν οι στήλες των προϊόντων (*p1,..,p10*) μαζί με τη στήλη του μέσου όρου των εσόδων από κάθε υποσύνολο προϊόντων που αντιστοιχεί σε κάθε γραμμή (*average_income*).

Σε πρώτο βήμα, εφαρμόστηκε κωδικοποίηση (encoding) με χρήση της συνάρτησης *get_dummies*, στις στήλες των προϊόντων (*p1,..,p10*). Έπειτα, πραγματοποιήθηκε η τεχνική Split Data με διαχωρισμό σε train και test set, όπως στο πρώτο σύνολο δεδομένων, δηλαδή

80% και 20%, αντίστοιχα. εφαρμόστηκαν τα Regression μοντέλα μηχανικής μάθησης: Linear Regression, SVM, KNN και Neural Network. Στο παρακάτω πίνακα [Πίνακας 10] φαίνονται οι τιμές Score (r2 score) και MSE (Mean Square Error) για καθένα από αυτά.

	Linear Regression	SVM	KNN	Neural Network
Score	0.71	0.31	0.48	0.69
MSE	9552.64	23749.01	18497.83	10726.28

Πίνακας 10. Score και MSE των Regression μοντέλων



Εικόνα 15. Ραβδόγραμμα των τιμών Score για τα Regression μοντέλα

Β2) Προ-επεξεργασία και εφαρμογή Classification μοντέλων

Για συγκεκριμένο υποσύνολο *l* προϊόντων από τα *m*, ζητήθηκε αν τα αναμενόμενα έσοδα υπερβαίνουν ή όχι τη μέση τιμή αναμενόμενων εσόδων. Το ερώτημα αυτό δοκιμάστηκε με δύο τρόπους, όπως αναλύθηκε και στην περιγραφή για το πρώτο σύνολο δεδομένων.

Α Τρόπος (χωρίς encode για τις κατηγορικές τιμές των προϊόντων)

Σε πρώτο στάδιο αποφασίστηκε να χρησιμοποιηθεί το DataFrame με στήλες τα 10 προϊόντα (*p1,..,p10*) μαζί με τη μέση τιμή των εσόδων κάθε υποσυνόλου (*average_income*).

Αρχικά, έγινε διαχωρισμός στο σύνολο δεδομένων σε τιμές εισόδου–ανεξάρτητες μεταβλητές (*p1,..,p10*) και τιμή εξόδου-εξαρτημένη μεταβλητή (*average_income*) καθώς και σε training και test (80% - 20% αντίστοιχα). Στη συνέχεια, υπολογίσθηκε ο μέσος όρος των εσόδων του training set. Έπειτα, με κατάλληλες μετατροπές έγινε αλλαγή της Label τιμής σε 1 και 0 (υπερβαίνει-δεν υπερβαίνει το μέσο όρο).

Τέλος, εφαρμόστηκαν τα ακόλουθα τρία Classification μοντέλα μηχανικής μάθησης: Logistic Regression, KNN, SVM. Στο παρακάτω πίνακα [Πίνακας 11] καταγράφονται οι τιμές accuracy για αυτά.

Logistic Regression	KNN	SVM
61.56 %	70.77 %	61.56 %

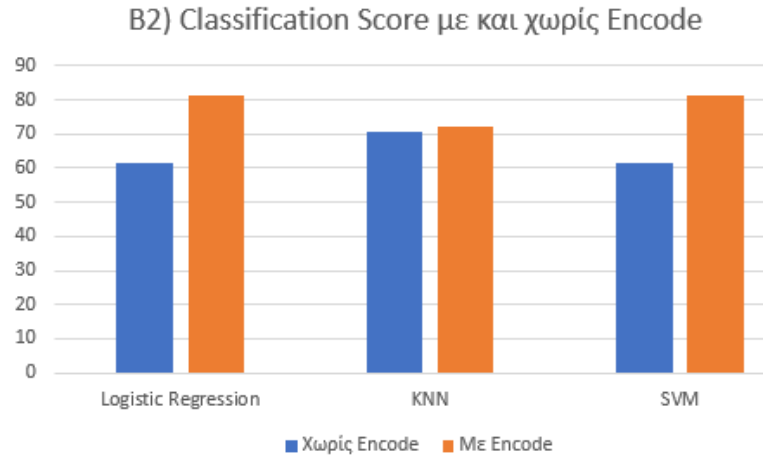
Πίνακας 11. Accuracy scores για τα Classification μοντέλα χωρίς Encode

Β Τρόπος (με encode για τις κατηγορικές τιμές των προϊόντων)

Το ερώτημα υλοποιήθηκε όπως και στον Α τρόπο με τη διαφορά ότι στις στήλες με τα προϊόντα εφαρμόστηκε η τεχνική κωδικοποίησης (encoding) με χρήση της συνάρτησης *get_dummies*. Στον παρακάτω πίνακα [Πίνακας 12] παρουσιάζονται οι τιμές accuracy για τα τρία Classification μοντέλα.

Logistic Regression	KNN	SVM
81.44 %	72.33 %	81.59 %

Πίνακας 12. Accuracy scores για τα Classification μοντέλα με Encode



Εικόνα 16. Ραβδόγραμμα με σύγκριση των Accuracy με και χωρίς encode

Σύγκριση των δύο Datasets

Όπως αναλύθηκε και επεξηγήθηκε στο κεφάλαιο της Περιγραφής, εφαρμόστηκαν τέσσερα Regression μοντέλα για τα ερωτήματα Α1 και Β1 και τρία Classification μοντέλα μηχανικής μάθησης για τα ερωτήματα Α2 και Β2. Ωστόσο εφαρμόζοντας τα μοντέλα και επιλέγοντας κάποιες μετρικές για την αξιολόγηση των μοντέλων, παρατηρήθηκε κάποια ουσιώδης διαφορά στην απόδοσή τους. Στο Κεφάλαιο αυτό αναλύεται η απόδοση των μοντέλων και γίνεται σύγκριση στα δύο σύνολα δεδομένων. Για τα ερωτήματα Α2 και Β2 επιλέχθηκαν οι μετρικές απόδοσης ως αποτελέσματα με τη χρήση κωδικοποίησης/encoding (Β Τρόπος), καθώς αποτέλεσε βοηθητική και στα δύο σύνολα δεδομένων.

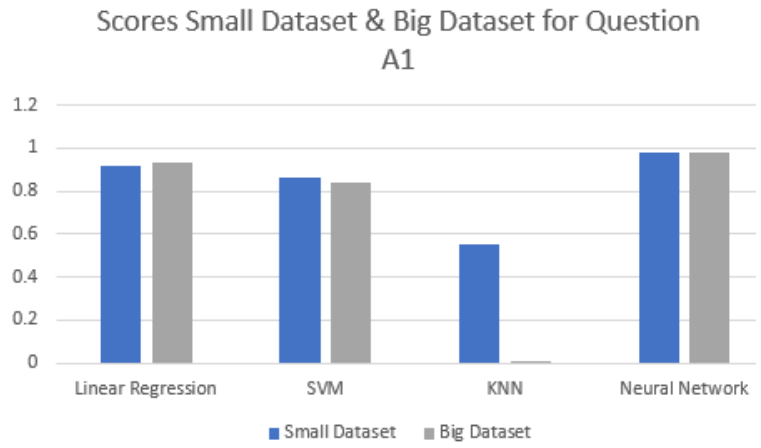
Για ευκολία και καλύτερη κατανόηση, το σύνολο δεδομένων με τα 17 προϊόντα θα αναφέρεται ως Small Dataset και το σύνολο δεδομένων με τα 20 προϊόντα θα αναφέρεται ως Big Dataset.

Σύγκριση για το Ερώτημα Α1

Όσον αφορά το Ερώτημα Α1, οι τιμές Score που καταγράφηκαν στα μοντέλα Linear Regression, SVM και Neural Network είναι αρκετά κοντινές με αποκλίσεις 0.01 και 0.02 τη μία υπέρ του μεγάλου συνόλου δεδομένων και την άλλη του μικρού συνόλου δεδομένων.

Αξιοσημείωτο, το γεγονός ότι κατά εφαρμογή του KNN καταγράφηκε Score ίσο με 0.55 στο μικρό σύνολο δεδομένων και από την άλλη, Score ίσο με 0.01 για το μεγάλο σύνολο

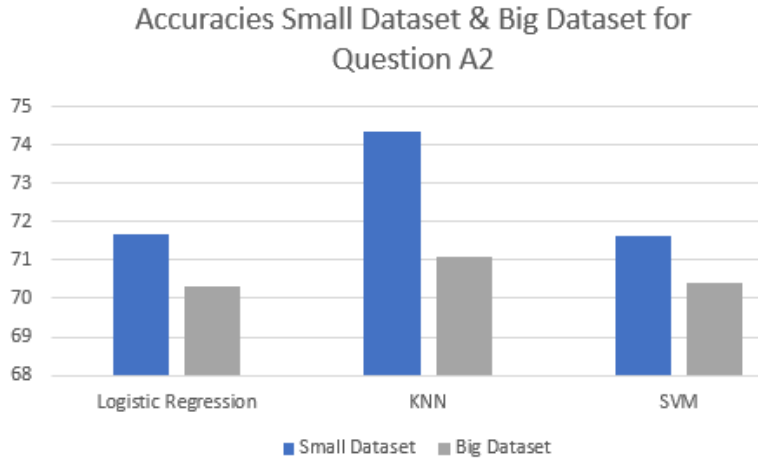
δεδομένων. Τα αποτελέσματα των συγκρίσεων απεικονίζεται στο παρακάτω ραβδόγραμμα [Εικόνα 17].



Εικόνα 17. Ραβδόγραμμα με τις τιμές Score για το Big και Small Dataset

Σύγκριση για το Ερώτημα Α2

Όσον αφορά το Ερώτημα Α2, που σχετίζεται την εφαρμογή των Classification μοντέλων, οι τιμές accuracy για τα στα δύο σύνολα δεδομένων φαίνεται να έχουν κάποιες διαφορές. Και στα τρία μοντέλα (Logistic Regression, KNN, SVM) η διαφορά ακρίβειας των δύο dataset είναι περίπου μεταξύ 1.5 έως 3.5%, με μεγαλύτερο accuracy για το Small Dataset. Σε γενικές γραμμές τα μοντέλα φαίνονται αποδοτικά. Στο παρακάτω ραβδόγραμμα [Εικόνα 18] παρουσιάζονται οι τιμές accuracy για τα τρία μοντέλα των δύο συνόλων δεδομένων.



Εικόνα 18. Ραβδόγραμμα με τις τιμές Accuracy για το Big και Small Dataset

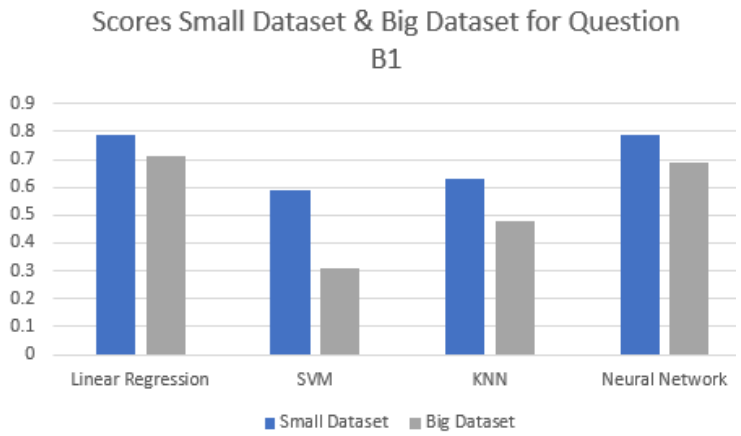
Σύγκριση για το Ερώτημα B1

	Linear Regression	SVM	KNN	Neural Network
Small Dataset Score	0.79	0.59	0.63	0.79
Big Dataset Score	0.71	0.31	0.48	0.69

Πίνακας 13. Τιμές Score στα 4 μοντέλα για το Small και Big Dataset

Όσον αφορά το Ερώτημα B1, που σχετίζεται με την εφαρμογή των Regression μοντέλων, παρατηρήθηκε ότι υπάρχουν σημαντικές διαφορές στις τιμές Score μεταξύ των δύο συνόλων δεδομένων. Συγκεκριμένα, η μικρότερη διαφορά παρατηρήθηκε ίση με 0.08 για το Linear Regression, με καλύτερο ποσοστό απόδοσης για το μικρό σύνολο δεδομένων. Η μεγαλύτερη διαφορά παρατηρήθηκε ίση με 0.28 για το SVM.

Συνολικά, σε όλα τα Regression μοντέλα μεγαλύτερο Score καταγράφηκαν στο μικρό σύνολο δεδομένων. Στο παρακάτω ραβδόγραμμα [Εικόνα 19] παρουσιάζονται οι τιμές Score για τα δύο σύνολα δεδομένων.



Εικόνα 19. Ραβδόγραμμα με τις τιμές Score για το Big και Small Dataset

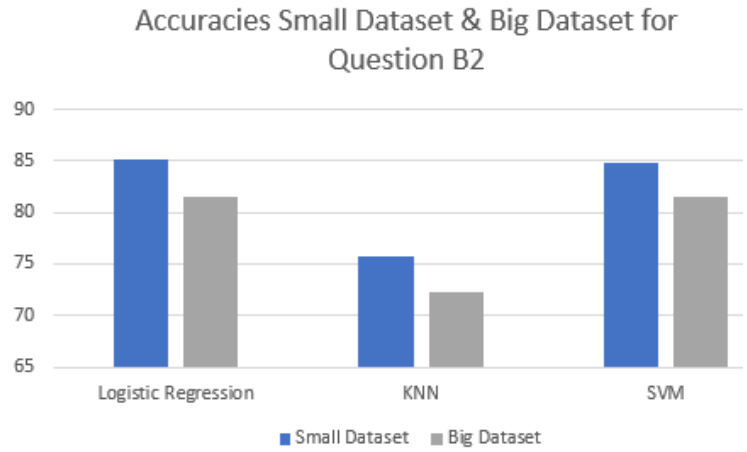
Σύγκριση για το Ερώτημα B2

	Logistic Regression	KNN	SVM
Small Dataset Accuracy	85.08 %	75.78 %	84.83 %
Big Dataset Accuracy	81.44 %	72.33 %	81.59 %

Πίνακας 14. Τιμές Accuracy στα τα 3 μοντέλα για το Small και Big Dataset

Για το ερώτημα B2, με την εφαρμογή των Classification μοντέλων, παρατηρήθηκαν σε γενικές γραμμές καλύτερες αποδόσεις στο μικρό σύνολο δεδομένων από ότι στο μεγάλο. Η διαφορά αυτή στις τιμές Score φαίνεται να είναι περίπου ίση με 4% στο Logistic Regression

και περίπου 3% στα KNN και SVM. Στο παρακάτω ραβδόγραμμα [Εικόνα 20] απεικονίζονται τα αποτελέσματα αυτά.



Εικόνα 20. Ραβδόγραμμα με τις τιμές Accuracy για το Big και Small Dataset

Σύγκριση των MSE (Mean Square Error) για τα ερωτήματα A1 και B1 στο Small και Big Dataset

Όσον αφορά τις τιμές MSE (Mean Square Error), παρατηρήθηκε ότι στο μεγάλο σύνολο δεδομένων είναι κατά πολύ μεγαλύτερες σε σχέση με το μικρό σύνολο δεδομένων για τα αντίστοιχα Regression μοντέλα. Αυτό συμβαίνει λόγω της αύξησης της εξαρτημένης μεταβλητής και αντίστοιχα της εκτιμώμενης τιμής. Με την αύξηση της εξαρτημένης μεταβλητής Y, αυξάνονται οι διακυμάνσεις των εκτιμώμενων και των πραγματικών τιμών Y. Ως συνέπεια των παραπάνω, αυξάνονται τα τετράγωνα των διαφορών των προβλεπόμενων από των πραγματικών τιμών και έτσι, αυξάνεται και η τιμή MSE, δικαιολογώντας έτσι τα αποτελέσματα. Οι συγκρίσεις των τιμών MSE για τα ερωτήματα και τα δύο σύνολα δεδομένων παρουσιάζονται στον παρακάτω πίνακα [Πίνακας 15].

	Linear Regression	**SVM**	**KNN**	**Neural Network**
Small Dataset Question A1	231.66	411.71	1349.96	41.28
Big Dataset Question A1	1987.42	4781.27	30879.87	549.53
Small Dataset Question B1	433.92	875.68	766.66	447.20
Big Dataset Question B1	9552.64	23749.01	18497.83	10726.28

Πίνακας 15. Τιμές MSE για τα Small και Big Dataset για τα ερωτήματα A1 και B1

Συμπεράσματα

Συνοψίζοντας, στην παρούσα εργασία εφαρμόστηκαν Regression και Classification μοντέλα μηχανικής μάθησης για τις προβλέψεις όπου τέθηκαν προς επίλυση. Στα ερωτήματα A1 και B1 εφαρμόστηκαν τα Regression μοντέλα Linear Regression, SVM (Support Vector Machine), KNN (K-Nearest Neighbors), Neural Network, και στα ερωτήματα A2 και B2 τα Classification μοντέλα Logistic Regression, KNN, SVM.

Στη πορεία της μεθοδολογίας που προσεγγίστηκε, και συγκεκριμένα στην προ-επεξεργασία των δεδομένων πριν την εφαρμογή των μοντέλων, σε πρώτη φάση που δεν εφαρμόστηκε κωδικοποίηση στις κατηγορικές τιμές των προϊόντων, η απόδοση των μοντέλων δεν ήταν αρκετά ικανοποιητική. Ενώ σε δεύτερο στάδιο όπου εφαρμόστηκε κωδικοποίηση (encoding) η απόδοση των μοντέλων αυξήθηκε σε κάποιο σημαντικό βαθμό. Συνεπώς, προέκυψε το συμπέρασμα ότι η διαδικασία της κωδικοποίησης βοηθάει σημαντικά τα μοντέλα για καλύτερη απόδοση.

Όσον αφορά τη σύγκριση των δύο συνόλων δεδομένων Small και Big Dataset, έπειτα από την ανάλυση τους, παρατηρήθηκε ότι οι αποδόσεις των μοντέλων και στη περίπτωση του Regression αλλά και στη περίπτωση του Classification, τα Score και τα Accuracy ήταν αυξημένα στο Small Dataset. Συνεπώς τα μοντέλα λειτουργούν περισσότερο αποδοτικά στο Small Dataset με τις μικρότερες διαστάσεις.

Επιπλέον, σε σύγκριση των δύο συνόλων δεδομένων, παρατηρήθηκε ότι στα ερωτήματα A1 και B1 οι τιμές MSE (Mean Squared Error), εμφάνισαν μεγάλες διαφορές. Πιο συγκεκριμένα, και στο ερώτημα A1 και στο B1, οι τιμές MSE ήταν αρκετά μεγαλύτερες στο Big Dataset. Αυτό συνέβη, διότι οι τιμές πρόβλεψης στο Big Dataset είναι μεγαλύτερες και κατά συνέπεια, υπήρξε αύξηση των τετραγώνων των διαφορών των προβλεπόμενων από των πραγματικών τιμών, και κατά συνέπεια του μέσου όρου τους.

Επιπρόσθετα, αξίζει να σημειωθεί ότι, ο χρόνος που απαιτήθηκε για τη εκτέλεση των μοντέλων στο Big Dataset ήταν μεγαλύτερος από ότι στο Small Dataset, καθώς τα μοντέλα είχαν να επεξεργαστούν περισσότερα χαρακτηριστικά, και συνεπώς με τον τρόπο αυτό υπήρξε αύξηση της πολυπλοκότητας.