

פרויקט גמר 5 יחידות לימוד

התמחות – תכנון ותכנות מערכות Deep Learning

Mobile phone price prediction

Mobile price prediction

שם בית הספר: מקיף י"א ראשונים, ראשון לציון

שם העבודה: קביעת מחיר לטלפונים

שם התלמיד: בן אדזיאשבילי

ת.ז. התלמיד: 213217987

שם המנחה: דינה קראוס

תאריך ההגשה: 16.08.2021

התמחות – תכנון ותכנות מערכות Deep Learning

Mobile phone price prediction

בן אדדיאשבילי

פרויקט גמר 5 יחידות לימוד

התמחות – תכנון ותכנות מערכות Deep Learning

Mobile phone price prediction

תוכן עניינים

2.....	מבוא:
3.....	מבוא לקוד:
3.....	זרימת המידע בין יחידות:
4.....	מדריך למשתמש
8.....	בסיס הנתונים
9.....	מדריך למפתח
9.....	קובץ final.code:
12.....	קובץ KNN:
14.....	קובץ training:
16.....	הרצת התוכנה:
20.....	סיכום אישי/ רפלקציה
21.....	ביבליוגרפיה
21.....	נספחים

מבוא:

השנה אני וכיתתי התבקשנו לבצע ולהכין את פרויקט הגמר שלנו בנושא "deep learning", שבאמצעותו המחשב מדמה באופן מלאכותי את יכולת הלמידה של בני האדם. המוח מורכב מניורונים שקולטים את המידע מתאי עצב, מה שנותן לבני האדם את האפשרות ללמוד ולזכור דברים חדשים. באופן דומה בעזרת קוד ניתן לבנות למחשב רשת ניורונים מלאכותית, שבאמצעותה נוכל לדמות את תהליך הלמידה של בני אדם, ולהעניק "מוח" למחשב.

בחרתי לכתוב את הפרויקט שלי בעקבות רעיון שצץ במוחי בזמן שחיפשתי טלפון חדש בשוק, עשיתי הרבה השוואות מחירים, בדקתי נתונים על הטלפון, אך לא ידעתי כיצד לשפוט אם המחיר של הטלפון מתאים ליכולות ולנתונים שלו. לאחר שעות של מחקר מייאש, באינטרנט חשבתי על הרעיון לעשות בנושא הזה את פרויקט הגמר שלי, שמטרתו היא ללמד את המחשב לתמחר את הטלפון, בהתבסס על הנתונים features של הטלפון. מצאתי את הנתונים החשובים ביותר בנוגע לכל טלפון, כמו לדוגמה WiFi, 4G, גודל רוחב ועובי ועוד. בשל מחסור בנתונים והאמצעים לכתוב תוכנה שתיתן את המחיר המדויק של הטלפון, התוכנה מתמחרת את הטלפון מטווח של 0-3 כאשר אפס הוא מחיר זול מאוד, ושלוש הוא מחיר יקר מאוד. ובעתיד עם האמצעים והנתונים ההולמים והמתאימים, יהיה ניתן להפוך את הקוד לאפליקציה מתוחכמת יותר, שתהיה יעילה לכל צרכן שמחפש טלפון, ותעזור למנוע מחברות למכור טלפונים במחירים לא הולמים.

בסקירת המצב הקיים בשוק ישנם תוכנות שונות המבוססות של DL (Deep Learning) אשר נועדו לחזות מחיר של טלפון לפי תכונות שונות. השדרוג שמוצע בתוכנה שלי הוא העברת המחירים שחזתה התוכנה לטבלת אקסל כתלות ב 20 המשתנים. בנוסף בפרויקט שלי אני משתמש באלגוריתם KNN שמסתמך על דמיון בין אובייקטים שונים בdataset כמו לדוגמה שתי טלפונים שרוב הנתונים שלהם זהים, יהיו כנראה באותו במחיר. באמצעות KNN נשתמש ב dataset בקובץ train על מנת ללמד את המחשב לחזות מחירים בצורה נכונה. אלגוריתם זה מוכר ברמת הדיוק הגבוהה שלו בניתוח כמויות גדולות של data, במקרה שלנו 2000 מקרים.

במהלך הפרויקט נתקלתי בקשיים רבים, אחד מהם היה חוסר הניסיון בשפת התכנות (python) שהייתה חדשה לי כשהתחלתי, וכמובן שחוסר הניסיון שלי הקשה מאוד על כתיבת הקוד עקב שגיאות כתיב ומחסור בידע.

בנוסף על כך, האתגר הגדול ביותר שהיה לי במהלך הפרויקט הוא למידת נושא Deep Learning, על מה הוא מתבסס, איך הוא עובד, וכמובן איך להשתמש בו בקוד, ואת כל זה התבקשתי לעשות בנושא חדש לגמרי, שלא היה לי שום ידע קודם ורקע בו. בעזרת המורה שלי, אתרי אינטרנט ופורומים קהילתיים כמו stack overflow של מתכנתים מקצוענים וחובבנים, התמצאתי בספריות השונות ובתחום Deep Learning באופן כללי.

והאתגר השלישי שהייתי רוצה לציין, היה בחירת מודל האימון המתאים. בתחום ה Deep Learning ישנם מודלים רבים בהם משתמשים לאמן וללמד את המחשב, ויש מגוון רחב מאוד של מודלים לשימוש, ובגלל שהמגוון כל כך רחב היה קשה מאוד להחליט איזה מהמודלים יהיה הכי טוב ומתאים לפרויקט שלי. לאחר סקירה מעמיקה באינטרנט לקחתי ארבעה מודלים שלפי מה שקראתי עליהם, הם נשמעו מתאימים לפרויקט שלי, ולאחר שכתבתי 4 קודים בנפרד (אחד לכל פרוייקט) בחרתי את הקוד עם accuracy הגבוה ביותר מבין הארבעה, שהיה מודל KNN ובו בחרתי להשתמש בפרויקט הסופי שלי. מצאתי ולמדתי על המודל באתר towards science תחת הכותרת machine learning basics with the k-nearest neighbors.

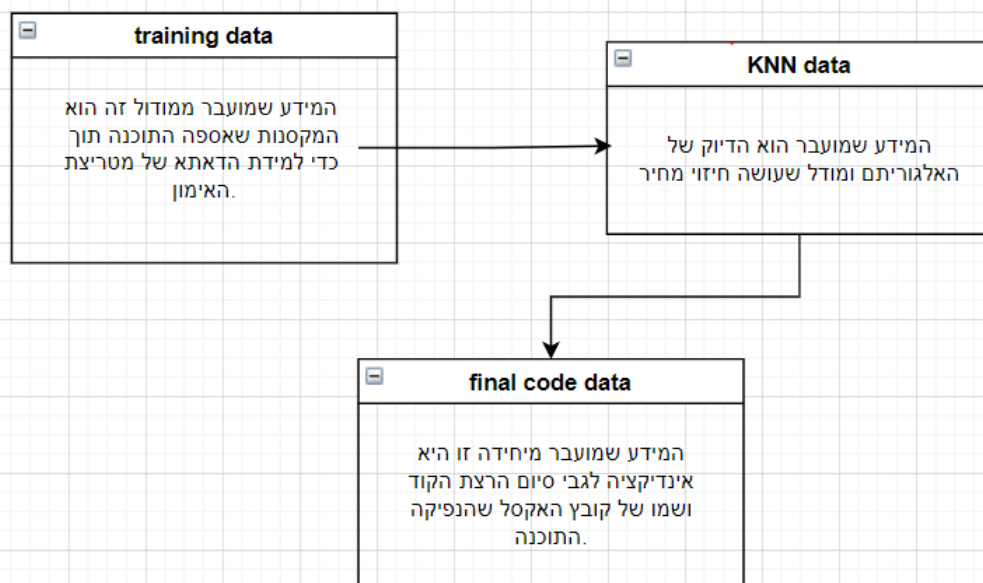
מבוא לקוד:

בקוד שלנו אנו רוצים לחזות מחיר של טלפון בהתאם ל20 תכונות שונות המשפיעות עליו.

אלה הן התכונות המשפיעות על מחיר הטלפון:

- ID - מייצג את מיקום הטלפון בטבלה..
- battery_power - האנרגיה המקסימלית שהבטרייה יכולה לאחסן ביחידת זמן הנמדדת ב mAh (אמפר לשעה).
- blue - מאפשר חיבור לבלוטות' או לא.
- clock_speed - המהירות בה המעבד מבצע פעולות.
- dual_sim - האם יש לו תמיכה של סים כפול או לא?
- fc - פיקסלים של מצלמה קידמית
- four_g - יש לו G4 או לא
- int_memory - זיכרון פנימי בג'יגה
- m_dep - עומק הנייד בס"מ
- mobile_wt - משקל הנייד
- n_cores - מספר הליבות שיש למעבד
- pc - פיקסלים של המצלמה הראשית
- px_height - רזולוציית הפיקסלים לגובה
- px_width - רזולוציית הפיקסלים לרוחב
- ram - גישה אקראית לזכרון במגה בי"תים
- sc_h - גובה מסך הטלפון בס"מ
- sc_w - רוחב מסך הטלפון בס"מ
- talk_time - הזמן הארוך ביותר שטלפון בטעינה מלאה יכול להחזיק בשיחה.
- three_g - האם יש לו G3 או לא
- touch_screen - האם יש לו טאץ' או לא
- wifi - האם יש לו חיבור לווי פיי או לא

זרימת המידע בין יחידות:



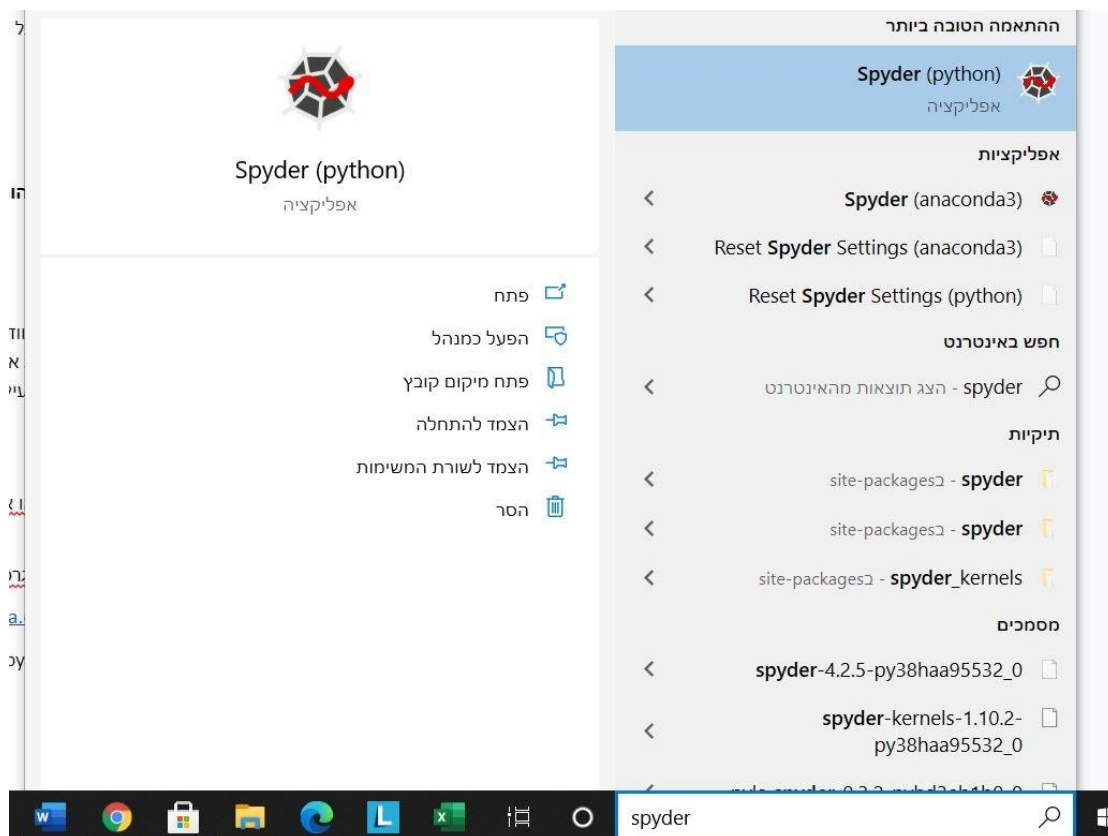
התמחות – תכנון ותכנות מערכות Deep Learning
Mobile phone price prediction
בן אדדיאשבילי

מדריך למשתמש

1. יש להתקין למחשב את סביבת העבודה Anaconda בגרסה 3.7 ומעלה .

-לינק להורדה <https://www.anaconda.com/products/individual>

2. לאחר מכן יש לחפש בתיבת החיפוש של המחשב spyder

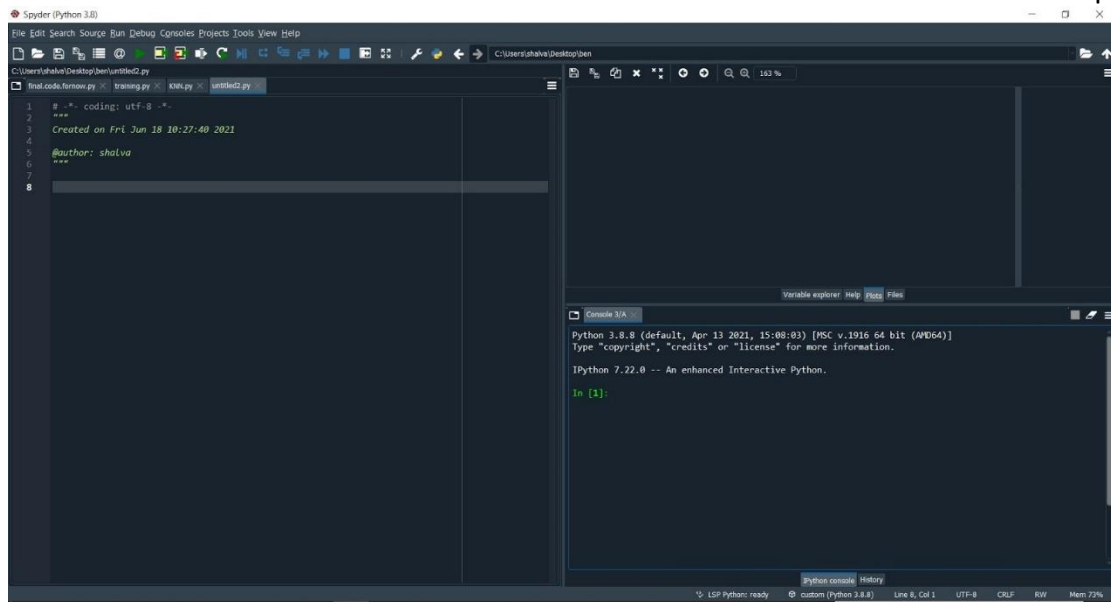


התמחות – תכנון ותכנות מערכות Deep Learning

Mobile phone price prediction

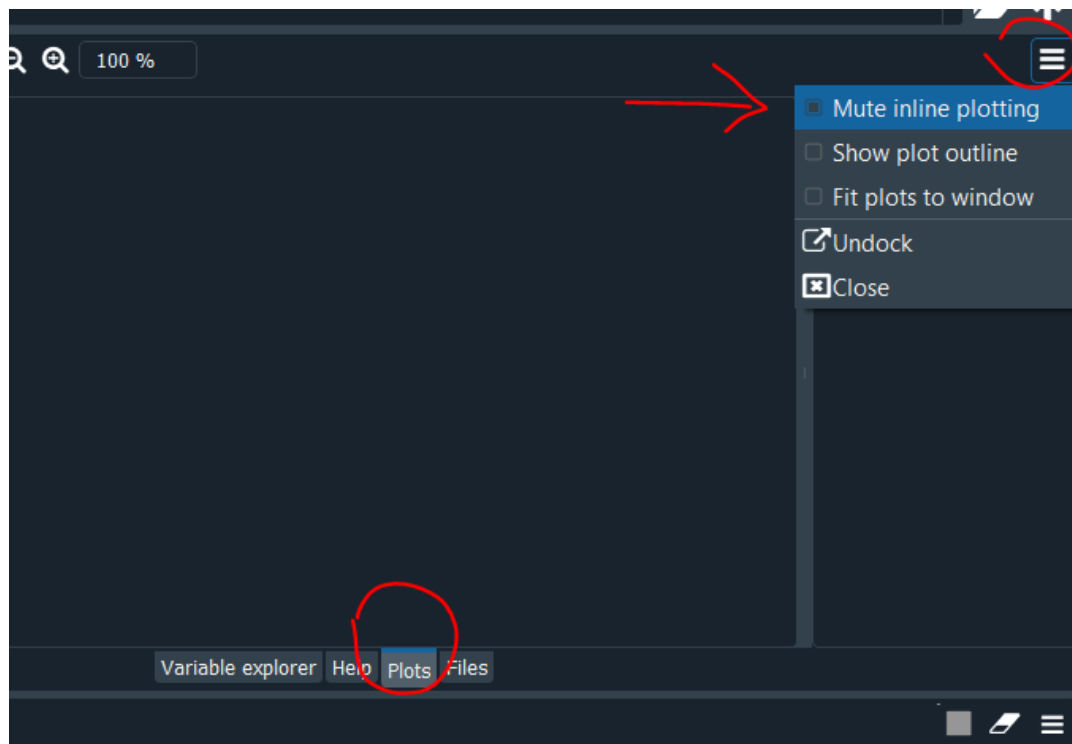
בן אדדיאשבילי

כך תראה התוכנה:



לשם הרצת קוד מסודרת על המריץ לשנות בחירה בהגדרות הפלוט (plots):

קודם נלחץ על מלבן הPlots לאחר מכן נפתח את ההגדרות (מסומן ב3 פסים אופקיים) ולבסוף נבחר את mute inline plotting ובטל בחירה מכל השאר.



3. יש להעלות כמה ספריות בהן התוכנה משתמשת:

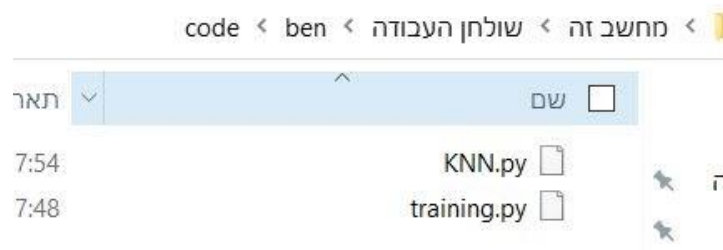
שם הספרייה	פקודת כתיבה
pandas	import pandas as pd
numpy	import numpy as np
sklearn.model_selection	from sklearn.model_selection import train_test_split
matplotlib.pyplot	import matplotlib.pyplot as plt
seaborn	import seaborn as sns
sklearn.neighbors	from sklearn.neighbors import KNeighborsClassifier
sklearn.metrics	from sklearn.metrics import accuracy_score
sklearn.model_selection	from sklearn.model_selection import GridSearchCV

4. בנוסף יש להוריד את המודולים הבאים:

שם המודול	צורת כתיבה
training	from training import x_train,y_train,x_valid,y_valid
KNN	from KNN import model_knn

שלי את הקבצים הבאים: GitHub ייש להוריד מן חשבון

- קבצי ה python עליהם מתבסס הקוד המודולים (final.code , training , KNN)
- על מנת להעלות את המודולים על הקבצים להשמר כקובצי פייתון py. דוגמא :

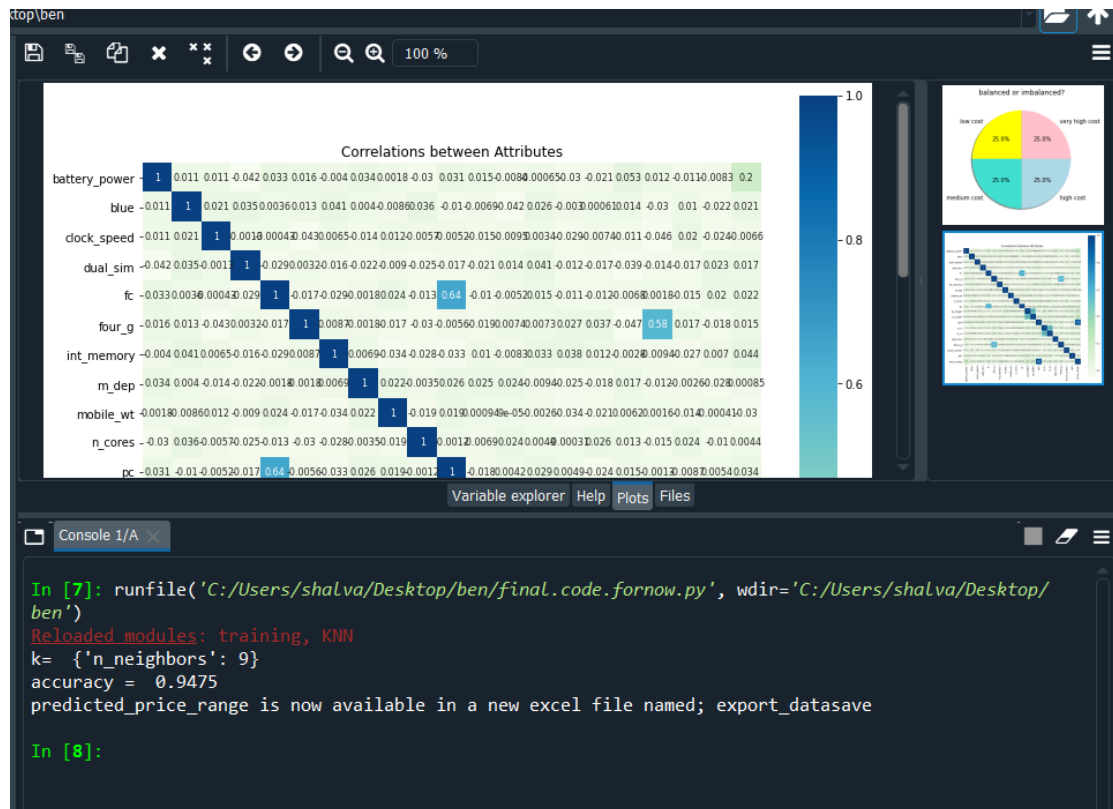


- את קובץ האקסל המכיל את מאגר המידע של הטסט. (test)
- את קובץ האקסל המכיל את מאגר המידע של לימוד המחשב (train)

התמחות – תכנון ותכנות מערכות Deep Learning Mobile phone price prediction בן אדדיאשבילי

6. אינדיקציה לכך שהתוכנה עובדת כפי שצריך:

על מנת לבדוק אם התוכנה הריצה את הקוד כפי שצריך, אל מסך הקלט פלט יודפס התצריך הבא:



יוצגו 2 תמונות אשר פותחו מתהליך הלמידה של התוכנה ובחלונית התחתונה תצוין פתיחת המודולים ולבסוף שורה המייצגת את סיום הליך ההרצה:

"predicted_price_range is now available in a new excel file named; export_datasave"

7. תפקידי היחידות בפרויקט:

- תפקיד המודול training הוא איסוף הנתונים של אימון התוכנה והפקת מסקנות כמו לדגומא מהי התוכנה בעלת ההשפעה הגבוהה ביותר על המחיר. הפלט של יחידה זו היא התמונות המייצגות את איזון המחירים והשפעתם של התכונות על מחיר הטלפון.
- תפקיד המודול KNN הוא ללמד את התוכנה ניתוח מחירי טלפונים בהתאם לתכונות שהוגדרו מראש באמצעות הנתונים והמסקנות שהתוכנה קיבלה ממודול training. הפלט של יחידה זו היא מידת הדיוק של השיטה והמחירים של כל אחד מהטלפונים.
- תפקיד היחידה final.code הוא להריץ בדיקה וניתוח מחירים על מטריצת datan של testn ולהפיק טבלת אקסל מן המטריצה ובנוסף להוסיף לקובץ עמודת מחיר חזוי.

התמחות – תכנון ותכנות מערכות Deep Learning Mobile phone price prediction בן אדדיאשבילי

בסיס הנתונים

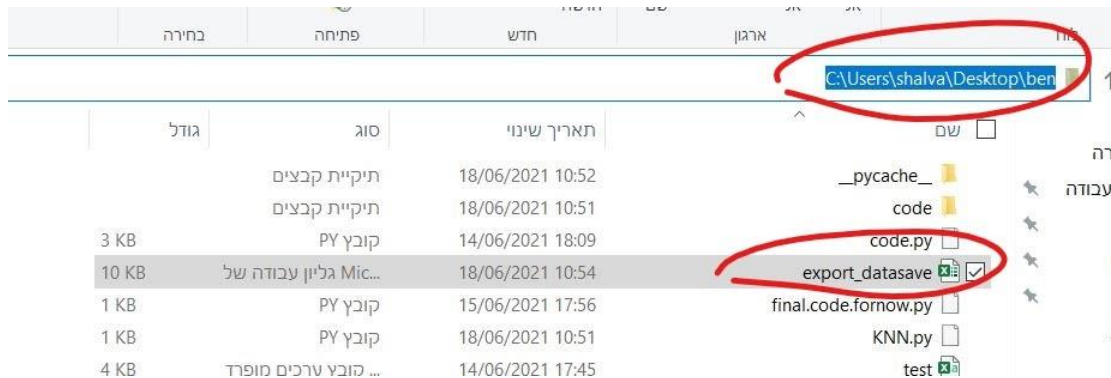
את תוצאות הטסט - התמחור שהתוכנה חישה לכל טלפון שנתנו לה לנתח אנו ממירים לקובץ אקסל הכולל את הנתונים של הטלפון והמחיר שנקבע לו.

ראשית הוספנו שורת קוד אשר מוסיפה את עמודת המחיר החזוי אל טבלת תכונות הטלפונים ולאחר מכן פיתחנו לקובץ אקסל את הטבלה שהמחשב יצר ושמרנו אותה תחת השם export_datasave.

```
# adds the predicted price range into the price_range columns
test_data['price_range'] = predicted_price_range

# exports the price range the computer calculated into a new excel file including all the data
df = pd.DataFrame (test_data)
df.to_excel ("C:/Users/shalva/Desktop/ben/export_datasave.xlsx" )
```

כאשר התוכנה מסיימת להריץ אץ הקוד, נוצר קובץ בשם ששמרנו אותו export_datasave התיקיה שהזנו.



וכאשר אנו פותחים את הקובץ אנו רואים כי נוספה לנו עמודה של מחיר והנתונים שהמחשב חזה

התמחות – תכנון ותכנות מערכות Deep Learning

Mobile phone price prediction

בן אדדיאשבילי

מדריך למפתח

C2		עריכה		תאים		סגנונות		ץ		מספר		ץ		יישור		ץ						
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
	battery pow	blue	lock speed dual sim	fc	four g	ht memor	m dep	mobile w	n cores	pc	px height	px width	ram	sc h	sc w	talk time	three g	uch scree	wifi	price range		
0	1043	1	1.8	1	14	0	5	0.1	193	3	16	226	1412	3000	12	7	2	0	1	0	2	
1	841	1	0.5	1	4	1	61	0.8	191	5	12	746	857	1900	6	0	7	1	0	0	1	
2	1807	1	2.8	0	1	0	27	0.9	186	3	4	1270	1366	2396	17	10	10	0	1	1	2	
3	1546	0	0.5	1	18	1	25	0.5	96	8	20	295	1752	3893	10	0	7	1	0	0	3	
4	1434	0	1.4	0	11	1	49	0.5	108	6	18	749	810	1773	15	8	7	1	0	1	1	
5	1464	1	2.9	1	5	1	50	0.8	198	8	9	569	939	3506	10	7	3	1	1	1	3	
6	1718	0	2.4	0	1	0	47	1	156	2	3	1283	1374	3873	14	2	10	0	0	0	3	
7	833	0	2.4	1	0	0	62	0.8	111	1	2	1312	1880	1495	7	2	18	0	1	1	1	
8	1111	1	2.9	1	9	1	25	0.6	101	5	19	556	876	3485	11	9	10	1	1	0	3	
9	1520	0	0.5	0	1	0	25	0.5	171	3	20	52	1009	651	6	0	5	1	0	1	0	
10	1500	0	2.2	0	2	0	55	0.6	80	7	6	503	1336	3866	13	7	20	0	1	0	3	
11	1343	0	2.9	0	2	1	34	0.8	171	3	6	235	1671	3911	15	8	8	1	1	1	3	
12	900	1	1.4	1	0	0	30	1	87	2	3	829	1893	439	6	2	20	1	0	0	0	
13	1190	1	2.2	1	5	0	19	0.9	158	5	15	227	1856	992	13	0	16	1	1	0	0	
14	630	0	1.8	0	8	1	51	0.9	193	8	9	1315	1323	2751	17	6	3	1	1	0	2	
15	1846	1	1	0	5	1	53	0.7	106	8	7	185	1832	563	9	5	10	1	0	1	0	
16	1985	0	0.5	1	14	1	26	1	163	2	17	613	1511	2083	13	3	14	1	1	0	2	

כל הקבצים נמצאים בתיקייה בשם ben ואלה הם הקבצים בהם נעשה שימוש:

C:\Users\shalva\Desktop\ben		שם	תאריך שינוי	סוג	גודל
		final.code.py	18/06/2021 12:55	קובץ PY	1 KB
		KNN.py	18/06/2021 10:51	קובץ PY	1 KB
		test	14/06/2021 17:45	קובץ ערכים מופרד	4 KB
		train	14/06/2021 10:20	קובץ ערכים מופרד	121 KB
		training.py	15/06/2021 17:48	קובץ PY	2 KB
		ספר פרויקט	14/06/2021 19:50	Microsof... מסמך של	71 KB

קובץ final.code:

בקובץ זה נמצא הקוד המוכן שבו נעשה שימוש במודולים וייצוא הטבלה הסופית אל קובץ אקסל.

תדפיס הקוד:

```
import pandas as pd
from KNN import model_knn

# importing test's data
test_data = pd.read_csv("C:/Users/shalva/Desktop/ben/test.csv")

# deleting the id columns
test_data=test_data.drop('id',axis=1)

# defining price range predicted ny KNN
predicted_price_range = model_knn.predict(test_data)

# adds the predicted price range into the price_range columns
test_data['price_range'] = predicted_price_range
```

```
# exports the price range the computer calculated into a new excel file including all
the data
df = pd.DataFrame (test_data)
df.to_excel ("C:/Users/shalva/Desktop/ben/export_datasave.xlsx" )

# done
print ("predicted_price_range is now available in a new excel file named;
export_datasave ")
```

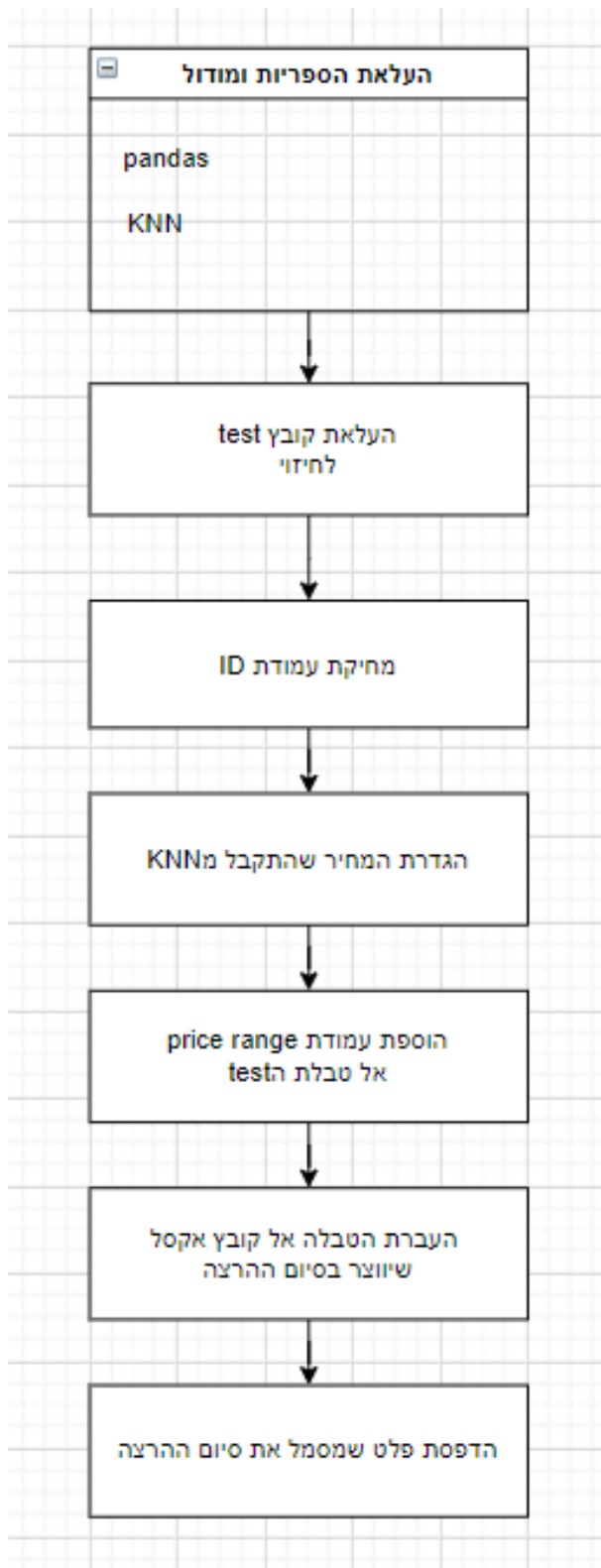
בקובץ זה אנו מעלים 2 ספריות:

pandas - ספרייה המאפשרת ניתוח מידע. בקוד זה- ניתוח קובץ הtest.

KNN - מודול אשר כולל את אלגוריתם KNN לחיזוי המחיר של הטלפונים בtest.

בקוד זה אנו מעלים את קובץ הtest הכולל 50 טלפונים עם תכונות שונות. לאחר מכן אנו מוחקים באמצעות התוכנה את עמודת הID מכיוון שבקובץ הtrain עליו בוצעה הלמידה של התוכנה, לא מופיעה עמודה זו. לאחר מכן המחיר החזוי מוגדר כמחיר שחזזה על ידי אלגוריתם KNN והתוכנה מוסיפה את הערכים של המחירים שהמחשב חזה אל טבלת הtest כעמודה של price range. כעת התוכנה מייצאת את הטבלה שיצרה עם ניתוח המחירים אל קובץ אקסל בשם export_datasave, שומרת אותו בתיקיית ben ומדפיסה פלט אשר מסמל את סיום הרצת התוכנה.

תרשים זרימה:



קובץ המודול של אלגוריתם KNN שבו מתבצע חיזוי המחיר של כל טלפון.

```
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
from training import x_train,y_train,x_valid,y_valid

# finds the optimum 'k'
from sklearn.model_selection import GridSearchCV
parameters = {'n_neighbors':np.arange(1,30)}
knn = KNeighborsClassifier()

# calculating KNN's accuracy
model = GridSearchCV(knn, parameters, cv=5)
model.fit(x_train, y_train)
print ( "k= " , model.best_params_)

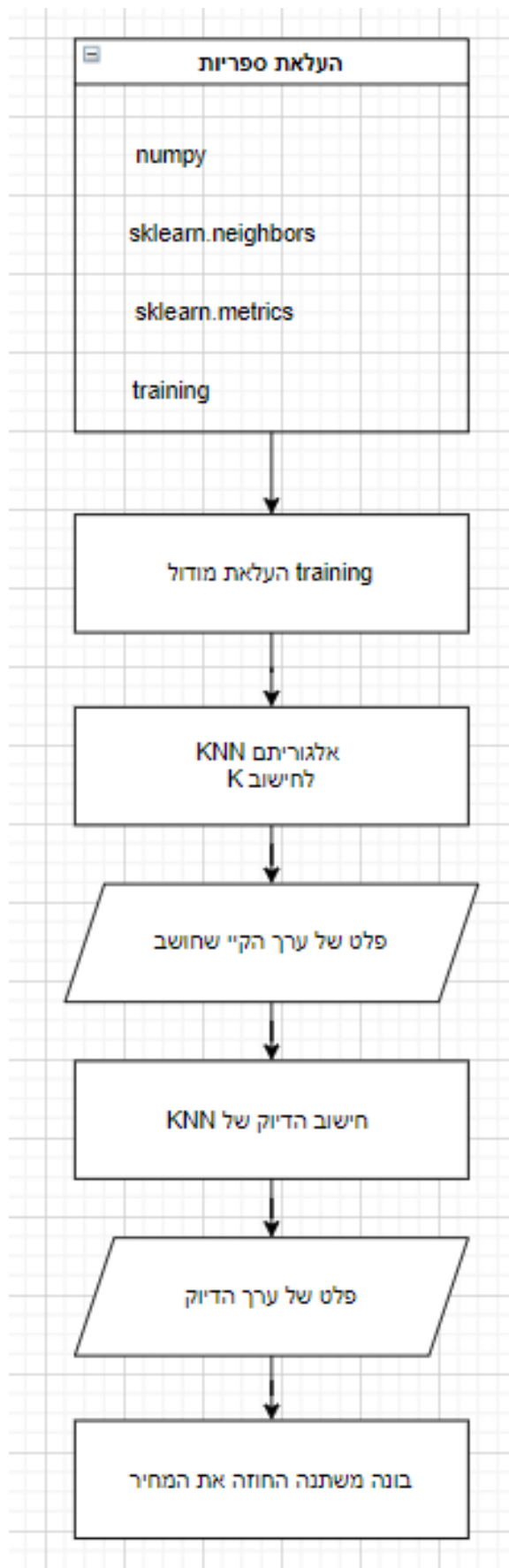
# uses the calculated 'k' to find accuracy and finds predicted price range
model_knn = KNeighborsClassifier(n_neighbors=9)
model_knn.fit(x_train, y_train)
y_pred_knn = model_knn.predict(x_valid)
acc_knn = accuracy_score(y_valid, y_pred_knn)
print ("accuracy = " , acc_knn)
```

בקוד זה אנו מעלים 5 ספריות:

numpy - ספרייה המאפשרת לעבוד עם מערכים רב מימדיים. במקרה שלנו מטריצת ה data של train ומטריצת ה data של test.
sklearn.neighbors - ספרייה יוצרת ומאמנת את אלגוריתם KNN.
sklearn.metrics - ספרייה המשמשת לחישוב הדיוק.
training - מודול אשר כולל את המידע והמסקנות שהמחשב ניתח מהמידע שקיבל מקובץ ה data של train.
sklearn.model_selection - ספרייה אשר משמשת למציאת ערך ה K באלגוריתם KNN.

בקוד זה אנו מוצאים את ערך ה K המיטבי ביותר לדיוק הגבוה ביותר. אחרי שמצאנו כי ה K=9 אנו מחשבים את הדיוק של האלגוריתם ומוצאים שהוא 0.9475 ולבסוף בעזרת חישוב ה K והדיוק התוכנה חוזרת את מחירי הטלפונים בהסתמך על הלמידה מה training.

תרשים זרימה



3. קובץ אקסל test: בקובץ זה נמצאים 50 טלפונים עם 20 תכונות שונות אך ללא מחיר. קובץ זה נועד להוות בקובץ החיזוי של התוכנה. התוכנה תחזה את מחיריהם של הטלפונים.
4. קובץ אקסל train : בקובץ זה נמצא מאגר מידע של 2000 טלפונים עם 20 תכונות שונות ומחיריהם. קובץ זה משמש ללימוד התוכנה ולחיזוי עתידי של טלפונים אחרים.

קובץ training:

הוא קובץ המודול שבו מתבצעת איסוף הנתונים מקובץ המידע שלנו (train) והפקת מסקנות בהתאם אשר יהוו שימוש לחיזוי עתידי.

```
import pandas as pd
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import seaborn as sns

# importing the train data

data = pd.read_csv("C:/Users/shalva/Desktop/ben/train.csv")

# data training the computer
y = data['price_range']
x = data.drop('price_range', axis = 1)
y.unique()
labels = ["low cost", "medium cost", "high cost", "very high cost"]
values = data['price_range'].value_counts().values
colors = ['yellow', 'turquoise', 'lightblue', 'pink']
fig1, ax1 = plt.subplots()
ax1.pie(values, labels=labels, colors=colors, autopct='%1.1f%%', shadow=True,
startangle=90)
ax1.set_title('balanced or imbalanced?')
plt.show()
x_train, x_valid, y_train, y_valid = train_test_split(x, y, test_size = 0.2,
random_state = 101, stratify = y)
fig = plt.subplots(figsize = (12, 12))
sns.heatmap(data.corr (), square = True, cbar = True, annot = True, cmap="GnBu",
annot_kws = {'size': 8})
plt.title('Correlations between Attributes')
plt.show ()
```

בקוד זה אנו מעלים 4 ספריות:

numpy - ספרייה המאפשרת לעבוד עם מערכים רב מימדיים. במקרה שלנו מטריצת data של
tarinn ומטריצת data של test.
sklearn.model_selection - ספרייה אשר משמשת למציאת ערך ה K באלגוריתם KNN.
matplotlib.pyplot - ספריית המשמשת ליצירת תמונות וגרפים ויזואליים.

התמחות – תכנון ותכנות מערכות Deep Learning

Mobile phone price prediction

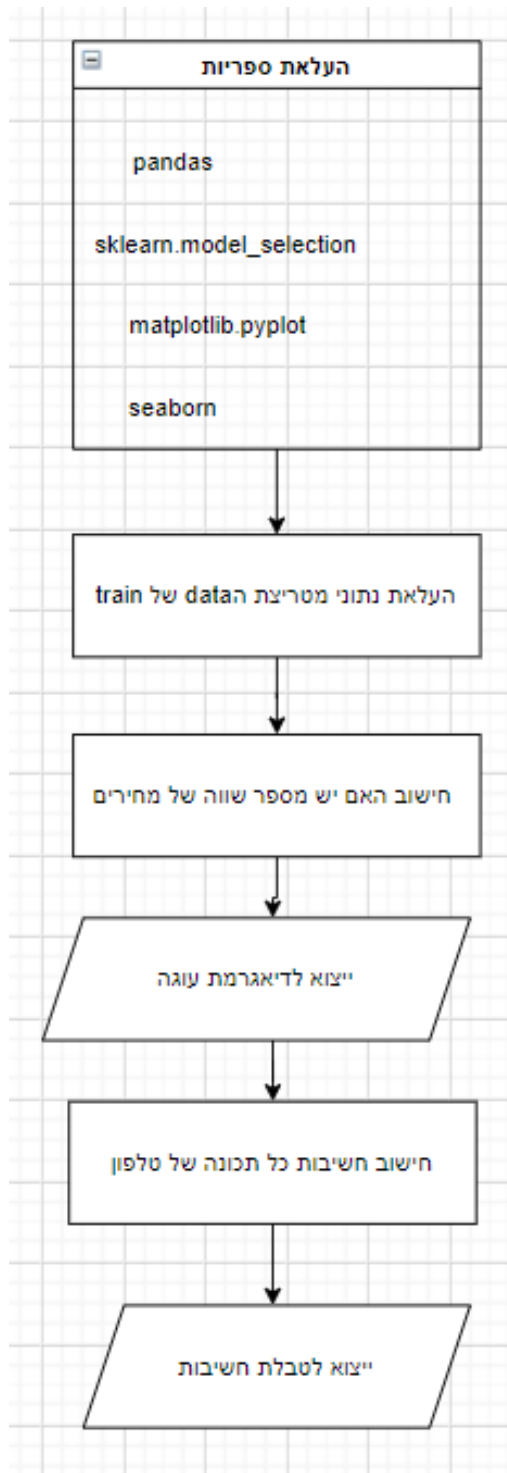
בן אדדיאשבילי

seaborn- ספרייה להמרת נתונים לגרפים ותמונות (הצגה ויזואלית).

בקוד זה אנו מעלים את קובץ train שלנו ומתחילים בללמד את התוכנה לחזות מחירים. ראשית אנו מוודאים כי כל אחד מערכי המחירים (מ-0 עד 3) מופיעים בצורה שווה בקובץ כך שהמערכת תוכל לנתח את הנתונים בצורה שווה ותקינה. לאחר מכן התוכנה מזהה את השפעות התוכנות על המחיר וקובעת את השפעתם.

בקוד זה אנו מייצאים שני תמונות אל מסך Plotsn. תמונה ראשונה היא האם מספר הפעמים שמופיע כל אחד מהמחירים מאוזן ושווה. והתמונה השנייה מציגה את השפעת כל תכונה על המחיר.

תרשים זרימה:



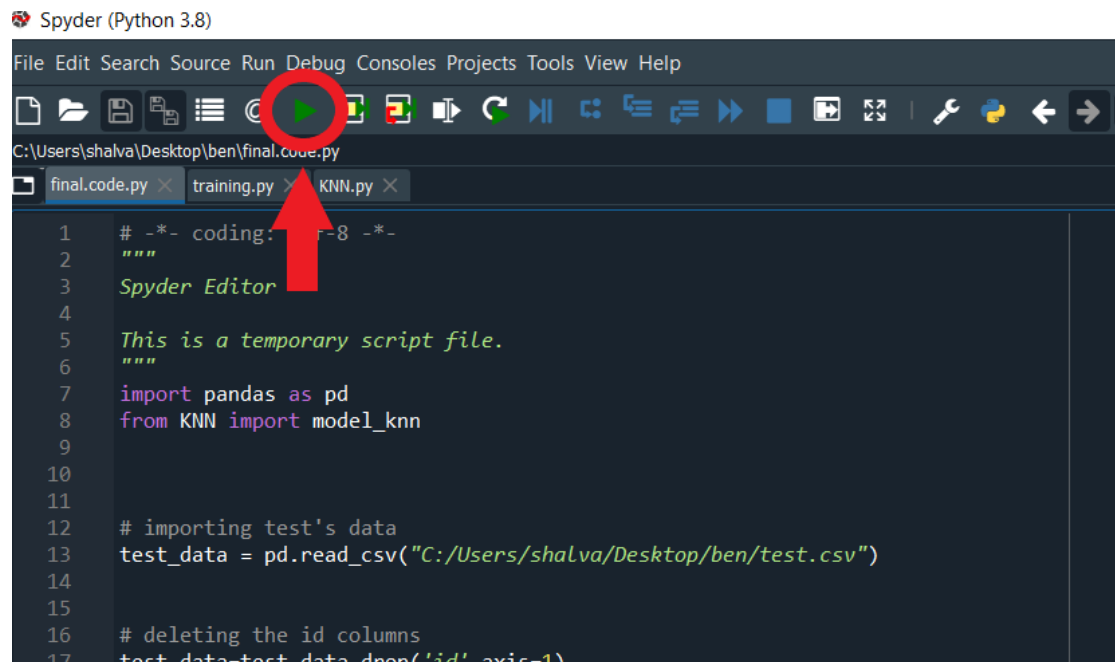
התמחות – תכנון ותכנות מערכות Deep Learning Mobile phone price prediction בן אדדיאשבילי

6. קובץ `export_datasave`: (לא מופיע בתמונה אך מוזכר בסעיפים אחרים) הוא הקובץ שאליו מוזנים המחירים שהתוכנה חוזרת לטלפונים.

הרצת התוכנה:

יש לפתוח בסביבת העבודה anaconda את קבצי המודולים: `training.py` ו-`KNN.py` ולאחר מכן להריץ את קובץ הפייתון: `final.code.py`.

על מנת להריץ את התוכנה יש ללחוץ על החץ הירוק:



הרצת קובץ זה תריץ את כל התכנית.

פלט ראשון: Reloaded modules: training, KNN

```
In [12]: runfile('C:/Users/shalva/Desktop/ben/final.code.py', wdir='C:/Users/shalva/Desktop/ben')
Reloaded modules: training, KNN
```

באשר התוכנית רצה במסך הקלט פלט תוצג העלאת המודולים (באדום).

קלט ראשון: `C:/Users/shalva/Desktop/ben/train.csv`

```
13
14 # importing the train data
15
16 data = pd.read_csv("C:/Users/shalva/Desktop/ben/train.csv")
17
18 # data training the computer
```

במודול `training` יש לנו קלט של ניתוח 2000 טלפונים עם מחירים המשמש כבלי ללימוד התוכנה.

ben מייצג את שם התיקייה בהם נמצאים כל הקבצים

Train מייצג את שם הקובץ ומועלה בקובץ csv (קובץ טקסט המאפשר לdata להישמר)

קלט שני: C:/Users/shalva/Desktop/ben/test.csv

```
# importing test's data
test_data = pd.read_csv("C:/Users/shalva/Desktop/ben/test.csv")
```

קלט זה נמצא בקוד הראשי ומטרתו להעלות את קובץ הטסט עליו יבוצע ניתוח המחירים.

ben מייצג את שם התיקייה בהם נמצאים כל הקבצים

test מייצג את שם הקובץ ומועלה בקובץ csv (קובץ טקסט המאפשר לdata להישמר)

קלט שלישי: C:/Users/shalva/Desktop/ben/export_datasave.xlsx

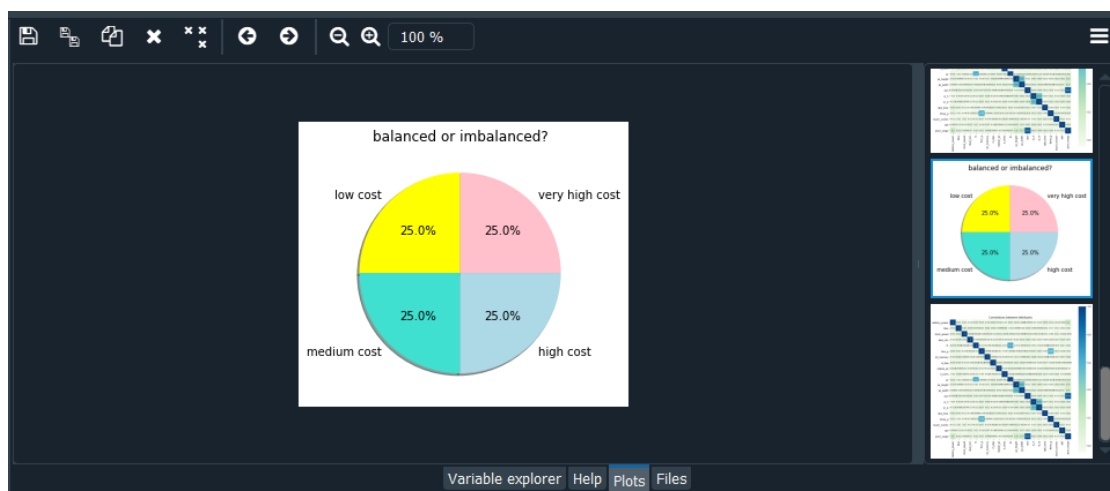
```
# exports the price range the computer calculated into a new excel file
df = pd.DataFrame (test_data)
df.to_excel ("C:/Users/shalva/Desktop/ben/export_datasave.xlsx" )
```

קלט זה נמצא בקוד הראשי ומטרתו לשמור את טבלת התכונות של הטלפונים כולל ניתוח המחיר שבוצע.

ben מייצג את שם התיקייה בהם נמצאים כל הקבצים

export_datasave מייצג את שם הקובץ ונשמר בקובץ xlsx (קובץ אקסל)

פלט שני: תמונה שהתוכנה יוצרת בה מצויין אחוז הפעמים שנותרו מספרי המחירים.



התמחות – תכנון ותכנות מערכות Deep Learning

Mobile phone price prediction

בן אדדיאשבילי

פלט זה נועד לבדוק אם מספר הפעמים שכל מחיר מ 0 עד 3 מוזכר יהיה שווה על מנת שהתוכנה תוכל ללמוד את הניתוח בצורה שווה ותלמד את כל טווחי המחירים במידה שתספיק לה להגיע לאחזו דיוק גבוה בכל טווח מחיר ותוכל לחזות את מחיר הטלפון.

פלט זה נמצא בplots

פלט שלישי: תמונה שהתוכנה יוצרת אשר מציגה את השפעת כל תכונה על מחיר הטלפון.



פלט זה נועד לבדוק את השפעת כל תכונה על המחיר וכך ללמוד לחזות את מחיר הטלפון הבא עם תכונות שונות בהתאם להשפעתם במחירי הtrain. זאת על מנת להראות למשתמש אילו תכונות הן המשמעותיות ביותר.

פלט זה נמצא בplots

פלט רביעי: $K = \{n_neighbors': 9\}$

```
In [12]: runfile('C:/Users/shalva/Desktop/ben/final.code.py', wdir='C:/Users/shalva/Desktop/ben')
k= {'n_neighbors': 9}
predicted_price_range is now available in a new excel file named; export_datasave
In [13]:
```

פלט זו מוצג על מסך הפלט קלט ומציין את ערך הK שחושב באלגוריתם KNN על מנת להגיע לדיוק מירבי.

פלט חמישי: accuracy = 0.9475

```
In [12]: runfile('C:/Users/shalva/Desktop/ben/final.code.py', wdir='C:/Users/shalva/Desktop/ben')
Reloaded modules: training, KNN
k= {'n_neighbors': 9}
accuracy = 0.9475
predicted_price_range is now available in a new excel file named; export_datasave
In [13]:
```

התמחות – תכנון ותכנות מערכות Deep Learning

Mobile phone price prediction

בן אדדיאשבילי

פלט זה מוצג על מסך הפלט קלט והוא מראה לנו את ערך הדיוק שהתוכנה הגיעה בלמידת הקוד באמצעות אלגוריתם KNN.

פלט שישי: predicted_price_range is now available in a new excel file named; export_datasave

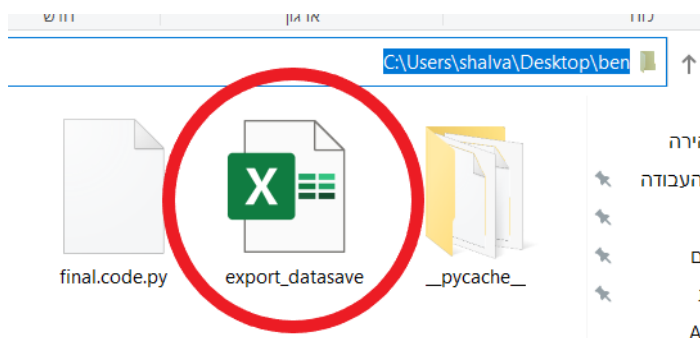
```
In [12]: runfile('C:/Users/shalva/Desktop/ben/final.code.py', wdir='C:/Users/shalva/Desktop/ben')
Reloaded modules: training, KNN
k= {'n_neighbors': 9}
accuracy = 0.9475
predicted_price_range is now available in a new excel file named; export_datasave
In [13]: |
```

פלט זה מציין את סיום הקוד ואומר לנו כי הטבלה של תכונות הטלפונים עם עמודת חיזוי המחיר נשמרה בקובץ excel בשם export_datasave.

בסיום הרצת הקוד יופיעו 2 התמונות בplots ובמסך הקלט פלט יודפסו כל הפלטים:

```
In [12]: runfile('C:/Users/shalva/Desktop/ben/final.code.py', wdir='C:/Users/shalva/Desktop/ben')
Reloaded modules: training, KNN
k= {'n_neighbors': 9}
accuracy = 0.9475
predicted_price_range is now available in a new excel file named; export_datasave
In [13]: |
```

ובתיקיה בה שמרנו את קובץ המחירים שחזתה התוכנה יופיע קובץ export_datasave, שיהיה זהה לקובץ test אך עם עמודה נוספת של price range.



התמחות – תכנון ותכנות מערכות Deep Learning
Mobile phone price prediction
בן אדדיאשבילי

סיכום אישי/ רפלקציה

העבודה על הפרוייקט הייתה מאתגרת אך מהנה עבורי, התמודדתי עם הרבה מצבים שבהם הרגשתי חסר אונים ושהקוד לעולם לא יעבוד בצורה שבה אני רוצה שיעבוד, וכשכן הצלחתי לתקן את מה שלא עבד, התחושה הייתה עילאית והיא זו שנתנה לי את המוטיבציה להמשיך להעמיק בנושא, לדבוק במטרה ולסיים את הפרוייקט בצורה הטובה ביותר.

הכלי הראשון שקיבלתי הוא החשיפה לשפת הפיתוח, שהייתה לי חדשה בתחילת העבודה, אך מהר מאוד נחשפתי למקורות מידע שונים שאפשרו לי ללמוד את כל הידע שהיה חסר לי על מנת לכתוב את הקוד. כלי נוסף שקיבלתי היא ההתנסות בתחום Deep Learning שהינו תחום מאוד מבוקש היום בתעשייה ולכיוונו העולם מתפתח (מכונות אוטונומיות, AI ועוד). לבסוף נחשפתי להרבה שיטות ומודלים שונים שבהם משתמשים בתחום Deep Learning, ונאלצתי לחקור על כל אחד בנפרד, לנסות לשלב אותו בפרוייקט שלי ולבחור את המודל שעבד והתאים בצורה הטובה ביותר לפרוייקט שלי.

הכלים שאני לוקח איתי להמשך הדרך הם הדבקות במטרה, וחוסר הרצון לוותר. היו הרבה רגעים במהלך כתיבת הקוד שגרמו לי להרגיש חסר אונים אבל זה לא מנע ממני מלהמשיך לחפש את הפתרון ואני חושב שאלו בדיוק התכונות שמתכנת טוב צריך שיהיה לו. בנוסף על כך למדתי להתמצאות באינטרנט ואיך לעשות חיפוש יעיל באינטרנט בנושאים ספציפיים, מה שיעזור לי בהמשך הדרך כמתכנת.

הקשיים שעמדו בפניי היו:

חוסר הנסיון שלי בשפת הפיתוח שעייב ותסכל אותי מאוד במהלך כתיבת הפרוייקט, כיוון שהיו שורות קוד שלא הצלחתי לכתוב כמו שצריך, מה שגרר זמן רב של debugging עד שירדתי לשורש הבעיות.

החשיפה לנושא מורכב ומסובך כמו Deep Learning ולמידת הנושא בעצמי, ללא עזרה חיצונית או מישהו שילמד אותי. זרקו אותי לתוך מאגר עצום של מודלים ושיטות והייתי צריך ללמוד ולהבין בעצמי במה להשתמש ואיך להשתמש.

המסקנות שלי הם שלפני שמתחילים לכתוב קוד בנושא חדש, עדיף לשבת ולקרוא על הנושא, כיצד הוא עובד ואיזה מודלים ואפשרויות הוא מכיל, לפני תחילת כתיבת הקוד. הבנתי שלקפוץ ישר לתוך כתיבת הקוד לא בהכרח יהיה יעיל יותר מאשר לחכות לקרוא ולהבין לפני כתיבת הקוד.

ולכן אם הייתי מתחיל היום את כתיבת הפרוייקט, הייתי יושב במשך כמה שעות טובות ומנסה לחקור לעומק כיצד עובד Deep Learning, על מה הוא מתבסס, ואיך ניתן להשתמש בו. ורק לאחר מכן הייתי מתחיל לחקור איזה מודל למידה יתאים לפרוייקט, לחפש פעולות וספריות מתאימות, ולהתחיל את כתיבת הקוד.

אם הייתי לומד על הרקע של Deep Learning, על מה הוא מתבסס, וכיצד הוא פועל, הייתי מבין איך המודלים השונים עובדים, ולמה אני צריך לכוון על מנת להשיג accuracy גבוה. ובכך הייתי חוסך לעצמי זמן רב של חיפושים וכתיבת קודים שונים לבדיקת accuracy.

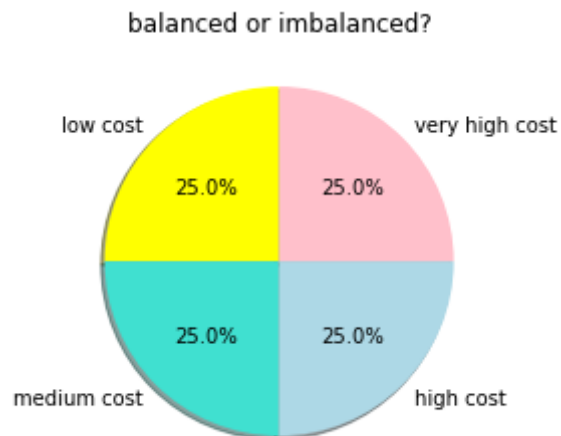
התמחות – תכנון ותכנות מערכות Deep Learning
Mobile phone price prediction
בן אדדיאשבילי

ביבליוגרפיה

- Vikramaditya Singh Bhati, 2018, Mobile Price Prediction
<https://www.kaggle.com/vikramb/mobile-price-prediction> אוחר מתוך prediction
- Abhishek Sharma, 2018, Mobile Price Classification
<https://www.kaggle.com/iabhishekoofficial/mobile-price-classification> אוחר מתוך classification (קבצי הדאטא של הtest והtrain)
- <https://stackoverflow.com/> - פורום קהילתי של מתכנתים

נספחים

באן התלמיד יכול להוסיף הסברים על הטכנולוגיות שנעשה בהם שימוש, או כל מידע שיכול להועיל לקורא העבודה

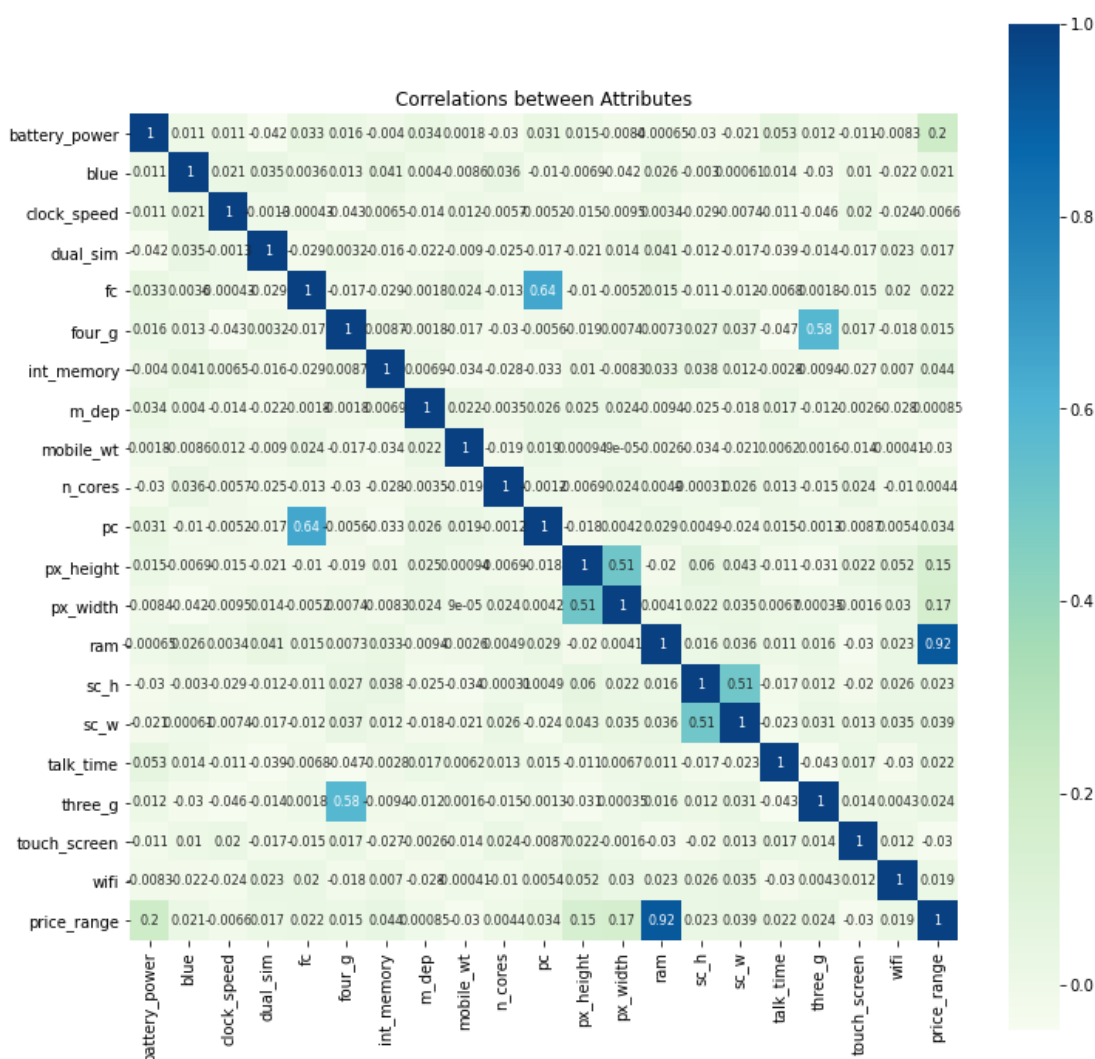


התמונה שנוצרת במודול training אשר מציגה לנו את האיזון בין מספר הפעמים שכל מחיר מופיע במטריצת datan של train.

התמחות – תכנון ותכנות מערכות Deep Learning

Mobile phone price prediction

בן אדדיאשבילי



גרף המייצג את הקשרים בין התכונות וההשפעה שלהם בקביעת מחיר הטלפון.

לפי עמודת הצבעים בצד ימין אנו יכולים לראות את רמת החשיבות של כל תכונה ורמת החשיבות בתיאום התכונות כלומר, כמות ההשפעה של 2 תכונות המוגדרות במספר גבוה בסולם שלהם על מחיר הטלפון.