# Mini project - Mobile Phones dataset

## Bar Gamliel, Boaz Bellomo

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

# part 1

## 1) תיאור מסד הנתונים שנבחר

In [2]:
```
original_df = pd.read_csv('phones_data.csv')
original_df = original_df.drop(columns=['Unnamed: 0']) #no need for doble indexing
original_df.head()
```

Out[2]:

| | brand_name | model_name | os | popularity | best_price | lowest_price | highest_price | sellers_a |
|---|---|---|---|---|---|---|---|---|
| **0** | ALCATEL | 1 1/8GB Bluish Black (5033D-2JALUAA) | Android | 422 | 1690.0 | 1529.0 | 1819.0 | |
| **1** | ALCATEL | 1 5033D 1/16GB Volcano Black (5033D-2LALUAF) | Android | 323 | 1803.0 | 1659.0 | 2489.0 | |
| **2** | ALCATEL | 1 5033D 1/16GB Volcano Black (5033D-2LALUAF) | Android | 299 | 1803.0 | 1659.0 | 2489.0 | |
| **3** | ALCATEL | 1 5033D 1/16GB Volcano Black (5033D-2LALUAF) | Android | 287 | 1803.0 | 1659.0 | 2489.0 | |
| **4** | Nokia | 1.3 1/16GB Charcoal | Android | 1047 | 1999.0 | NaN | NaN | |

המכשיר הסלולארי הפך להיות חלק אינטגרלי מחיי היום יום. אם בעבר רק חלק קטן מהאוכלוסיה היה מסתובב עם מכשיר טלפון נייד, היום מכשירים אלו הם נחלת הכלל. כמעט כל פעולה יומיומית דורשת גישה או שימוש במכשיר מסוג זה, ואף יש השלכות משפטיות וחוקיות לגישה למכשירים של אחרים. במרוצת השנים, סוגים שונים של מכשירים יצאו, חלקם היו פופולרים יותר וחלק פופולרים פחות. למכשירים שונים ישנם סוגי חומרה ותוכנה מגוונים, מחירים שונים ותפוצה שונה. בפרויקט זה נרצה לבחון מה תורם לפופולרות של מכישר מסויים. נבחן זאת באמצעות מסד

הנתונים שבחרנו המכיל מידע על טלפונים שהיו זמינים לרכישה באוקרינה. המסד מחזיק פרטים שונים על אותם טלפונים כגון הייצרן, תאריך יציאת הדגם, פרטים שונים על המחיר ועל רכיבי התוכנה והחומרה.

## 2) תיאור הפיצ'רים במסד הנתונים

ראיתי כי במקרים רבים ישנו תיאור של צבע המכשיר בתוך שם המודל שלו, נרצה לחלץ אותו ולהשתמש בו בזמן הניתוח. נציין, כי לכל צבע גוונים שונים בנתונים שלנו, אנחנו נתעלם מההבדל בגוונים.

In [3]:
```
# common color names
common_colors = ['Black', 'Blue', 'Pink', 'Red', 'Green', 'White', 'Gold', 'Silver

def extract_primary_color(model_name):
    for color in common_colors:
        if color.lower() in model_name.lower():
            return color
    return None

original_df['primary_color'] = original_df['model_name'].apply(extract_primary_col


original_df[['model_name', 'primary_color']].head(10)
```

Out[3]:

| | model_name | primary_color |
|---|---|---|
| **0** | 1 1/8GB Bluish Black (5033D-2JALUAA) | Black |
| **1** | 1 5033D 1/16GB Volcano Black (5033D-2LALUAF) | Black |
| **2** | 1 5033D 1/16GB Volcano Black (5033D-2LALUAF) | Black |
| **3** | 1 5033D 1/16GB Volcano Black (5033D-2LALUAF) | Black |
| **4** | 1.3 1/16GB Charcoal | Charcoal |
| **5** | 10 6/64GB Black | Black |
| **6** | 10 Lite 3/32GB Blue | Blue |
| **7** | 10 Lite 4/64GB Black | Black |
| **8** | 10 lite 3/128GB Blue | Blue |
| **9** | 10 lite 3/64GB Black | Black |

In [4]:
```
original_df.dtypes
```

Out[4]:
```
brand_name         object
model_name         object
os                 object
popularity          int64
best_price        float64
lowest_price      float64
highest_price     float64
sellers_amount      int64
screen_size       float64
memory_size       float64
battery_size      float64
release_date       object
primary_color      object
dtype: object
```

brand_name - קטגוריל. מתאר את שם היצרן

model_name - קטגוריל. מתאר את שם דגם המכשיר

os - קטגוריל. מתאר את סוג מערכת ההפעלה של המכשיר

popularity - נומרי. מדרג את הפופולריות של המכשיר מבין כל המכשירים במסד הנתונים

best_price - נומרי. אין הסבר מספק במקור המידע, אך ככל הנראה מתאר את המחיר המשתלם ביותר בשכלול נתונים אחרים

lowest_price - נומרי. המחיר הנמוך ביותר שהוצא עבור המכשיר במטבע אוקראיני

highest_price - נומרי. המחיר הגבוה ביותר שהוצא עבור המכשיר במטבע אוקראיני

sellers_amount - נומרי. כמות המוכרים אשר מוכרים את המכשיר באוקרינה

screen_size - נומרי. גודל המסך באינטשים

memory_size - נומרי. כמות הזיכרון בגיגה ביט

battery_size - נומרי. גודל הסלולה במילי אמפר לשעה

release_date - קטגוריל אורדינלי, מתאר את החודש בו יצא המכשיאר לשוק

primary_color - קטגוריל. מתאר את צבע המכשיר

## 3) תיאור כמות הנתונים במסד

In [5]:
```
num_duplicates = original_df['model_name'].duplicated().sum()
print(f"The number of duplicated entries based on the specified columns is {num_du
```

```
The number of duplicated entries based on the specified columns is 156
```

In [6]:
```
num_duplicates = original_df[['model_name', 'os', 'best_price', 'lowest_price', 'h
print(f"The number of duplicated considring all varubles is {num_duplicates}")
```

```
The number of duplicated considring all varubles is 156
```

ממעבר על הנתונים, ראינו כי קיימים כפילויות של דגמי מכשירים. יכולים להיות לכך מספר סיבות, לדוגמא הבדלים בין הדגמים שאינם מופעים במסד, או הטיית מדידה. אנחנו נניח כי קרתה פה הטיית מדידה, משום שערכי כל שאר הפיצ'ים זהים, למעט הפופולריות (שממוספרת ככמות המכשירים במסד). לכן נרצה למחוק את הכפילויות ולתת להן את ערך הפופולריות הממוצעת של כל קבוצת כפילויות.

In [7]:
```
mean_popularity = original_df.groupby('model_name')['popularity'].mean()
df = original_df.drop_duplicates(subset='model_name').copy()
df['popularity'] = df['model_name'].map(mean_popularity)
```

In [8]:
```
len(original_df)
```

Out[8]:
```
1224
```

In [9]:
```
len(df)
```

Out[9]:
```
1068
```

במסד הנתונים המקורי היו 1224 רשומות, לאחר הורדת הכפילויות נותרנו עם 1068 רשומות - דגמי טלפונים שונים

In [10]:
```
missing_values = df.isnull().sum()
missing_values
```

Out[10]:
```
brand_name          0
model_name          0
os                163
popularity          0
best_price          0
lowest_price      207
highest_price     207
sellers_amount      0
screen_size         2
memory_size        94
battery_size       10
release_date        0
primary_color      89
dtype: int64
```

קיימים מספר לא קטן של חוסרים בנתונים בפיצ'רים שונים. נחלק את הפיצ'רים עם החוסרים ל2 קטגרויות ונתמודד עם כל אחת מהן בנפרד - פיצ'רים עם חוסרים מועטים ופיצ'רים עם חוסרים רבים.

קיימות 2 רשומות בהן חסר גודל המסך ו10 רשומות בהן חסר גדול הסוללה. מספר הרשומות בהן המידע חסר הוא זניח לגדול מסד הנתונים ולכן נוכל להתעלם מהרשומות האלו מבלי לפגוע במחקר שלנו.

ברשומות של "מערכת הפעלה", "מחיר גבוה", "מחיר נמוך" ו"גודל זכרון" ישנם חוסרים רבים הנעים בין כ 10%-20% מהנתונים בכל אחד מהפיצ'רים האלו. ישנן מספר דרכים להתמודד עם החוסרים האלו:

1. התעלמות מהרשומות עם החוסרים. הבעיה עם השיטה הזו היא שמספר החוסרים הוא גדול ובמינימים נאבד כ20% מהנתונים שלנו
2. למלא את החוסרים בממוצע או החציון של העמודה שלהם. הבעיה עם השיטה הזו היא שחלק מהעמודות הן לא נומריות ולכן אין להן ממוצע או חציון, ואלו שכן נומריות, נראה בהמשך שההתפלגות היא לא נורמאלית ולכן השלמה זו תעוות את הנתונים שלנו.
3. למלא ערך רנדומלי בין הערך המינימלי או המקסימלי במקרה והמשתנה נומרי, או ערך רנדומלי לפי ההתפלגות אם המשתנה קטגוריל. שיטה זו בעייתית משום שהיא לא שומרת על היחסים בין המשתנים השונים (נראה מאוחר יותר כי קיימות קורולציות בין המשתנים) וגם קיימים ערכים קיצוניים להתפלגות, ולכן ערך רנדומלי בניהם לא ייצג נכונה את ההתפלגויות.

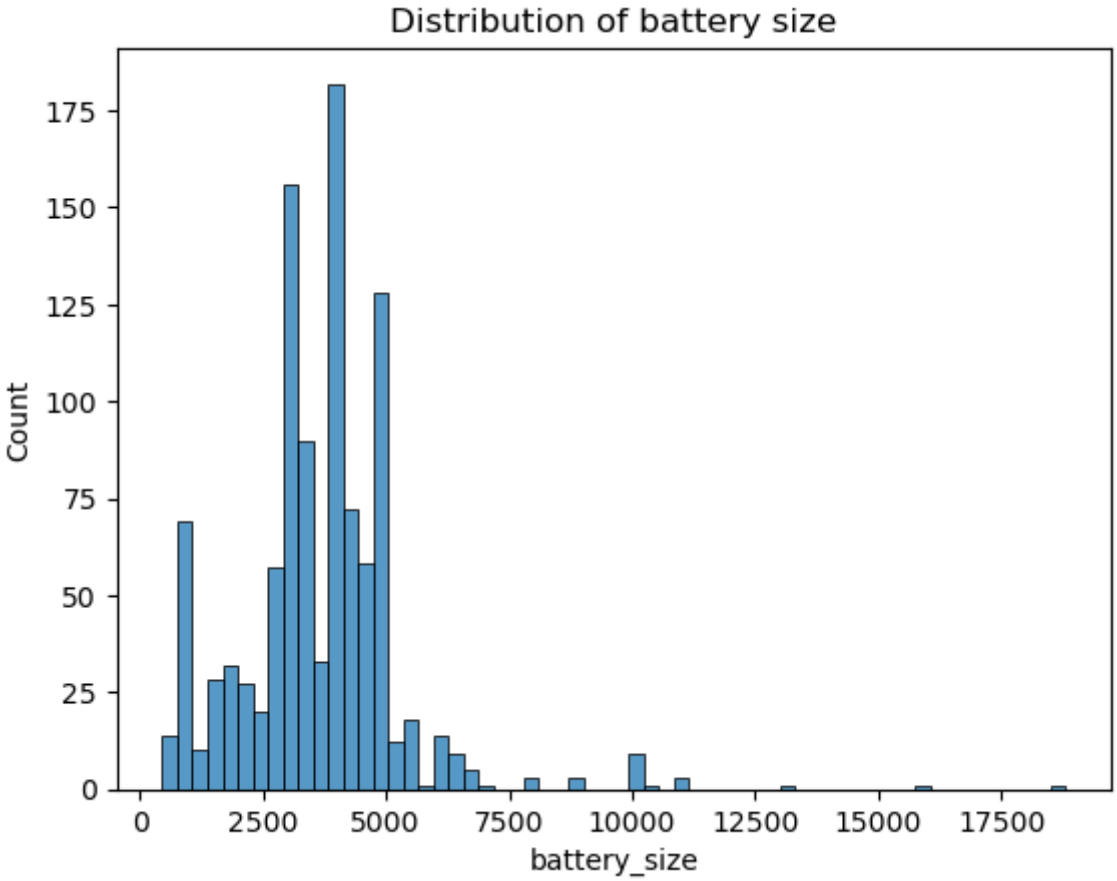
עקב הבעיות בשיטות השונות נשתמש בערכים שבהם יש מעט או אין בכלל חוסרים במקומות שניתן במקום אלו שיש בהם חוסרים רבים (כמו במחיר), ובמקומות שלא ניתן נקבל החלטות בהתאם לסוג הניתוח שנעשה.

# part 2

## 1) התפלגות נתונים

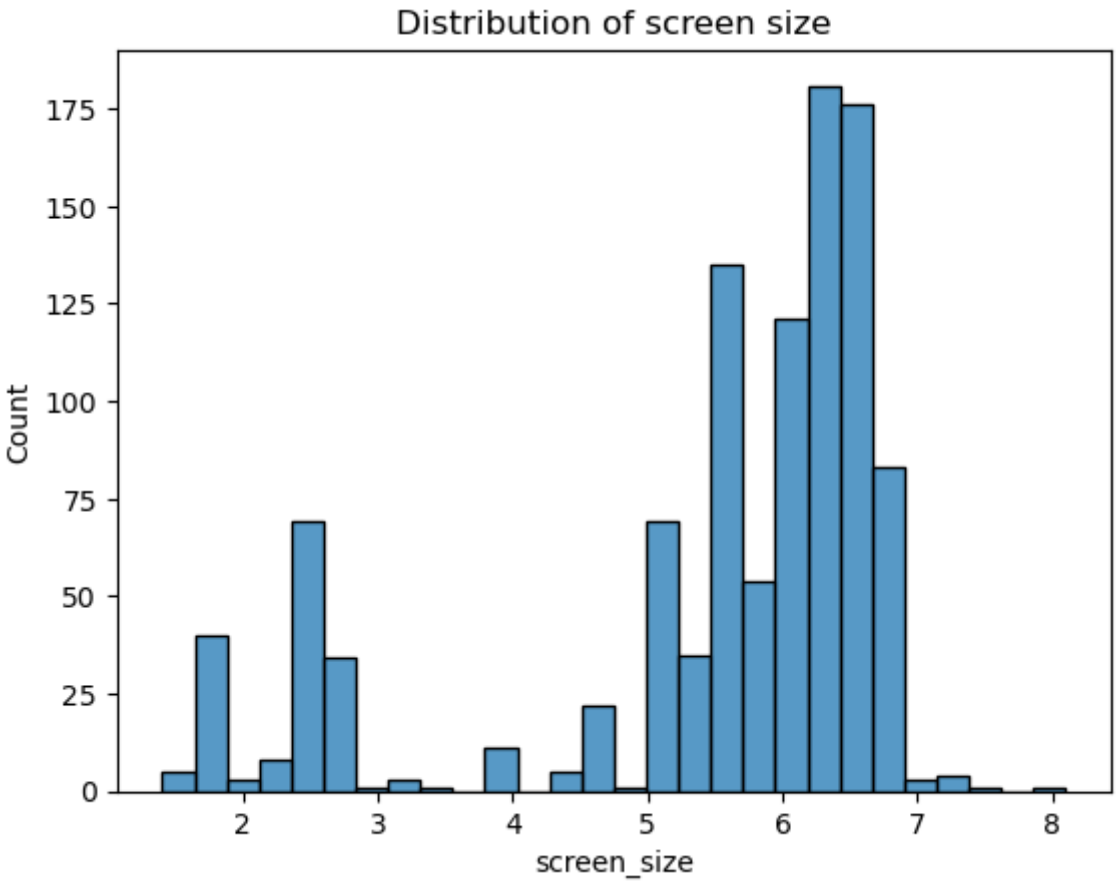
נרצה לראות באמצעות מדדים שונים אילו תכונות מאפיינות את רוב או חלק גדול מהמכשירים בנתונים שיש לנו. מאוחר יותר, נרצה לראות אם אכן התוכונת הנפוצות יותר משפיעות על הפופולריות של המכשיר.

In [11]:
```
sns.histplot(df['battery_size'].dropna())
plt.title('Distribution of battery size')
plt.show()

print('battery size mean is:',df['battery_size'].mean(), "and the median is:", df[
```




```
battery size mean is: 3656.4499054820417 and the median is: 3755.0
```

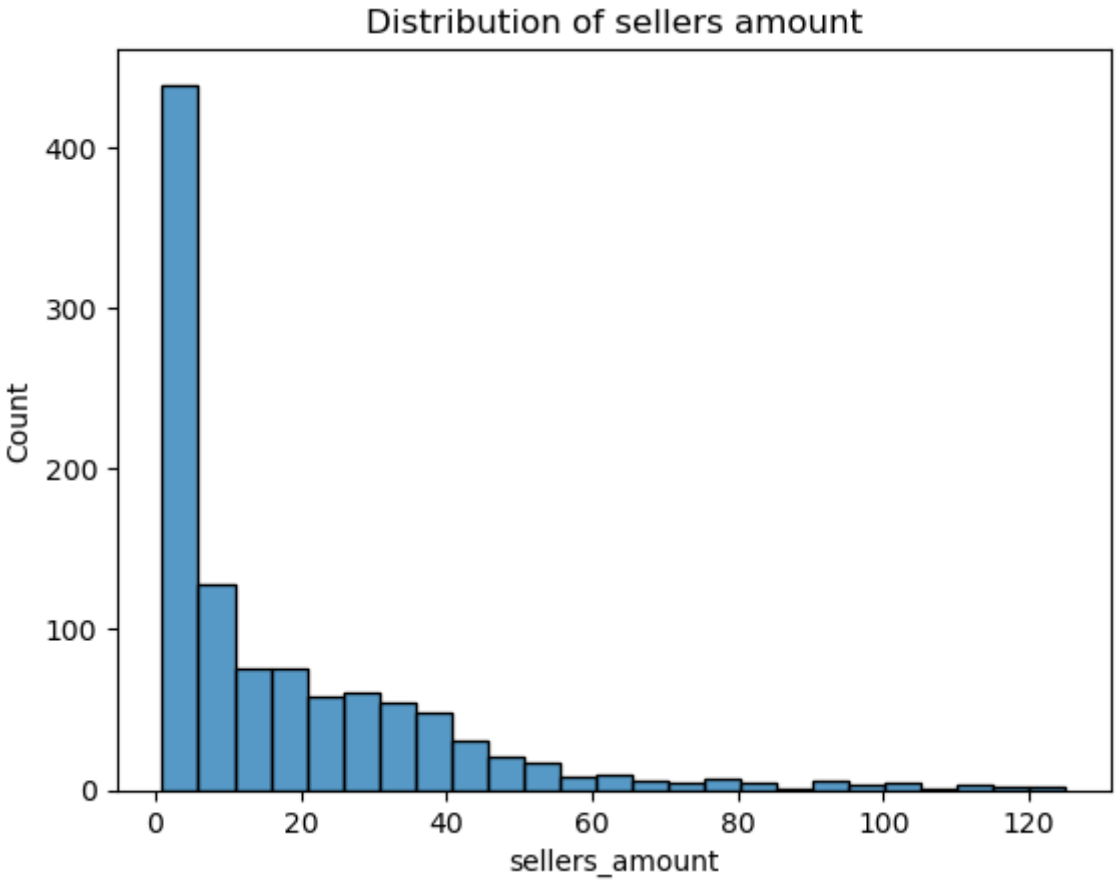
In [12]:
```
sns.histplot(df['screen_size'].dropna())
plt.title('Distribution of screen size')
plt.show()

print('screen size mean is:',df['screen_size'].mean(), "and the median is:", df['s
```




```
screen size mean is: 5.4460412757973735 and the median is: 6.0
```

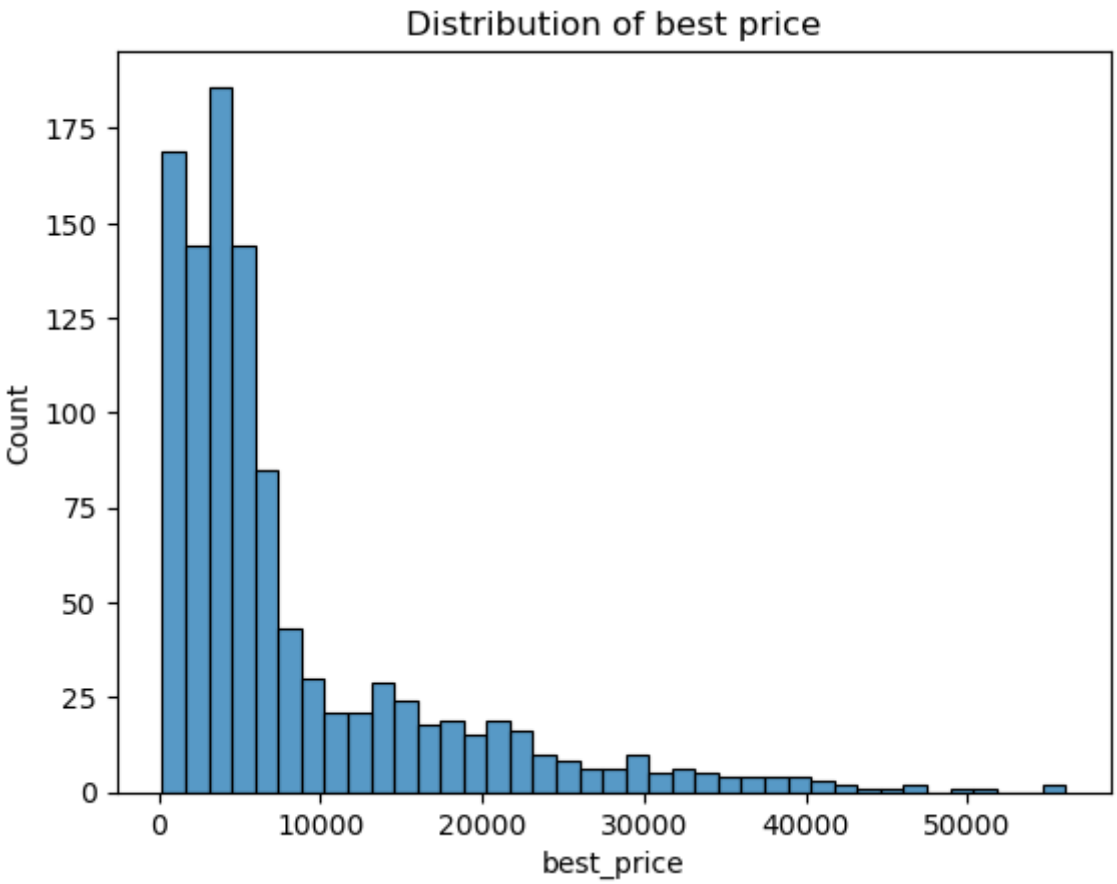
In [13]:

```
sns.histplot(df['sellers_amount'].dropna())
plt.title('Distribution of sellers amount')
plt.show()

print('sellers amount mean is:',df['sellers_amount'].mean(), "and the median is:",
```



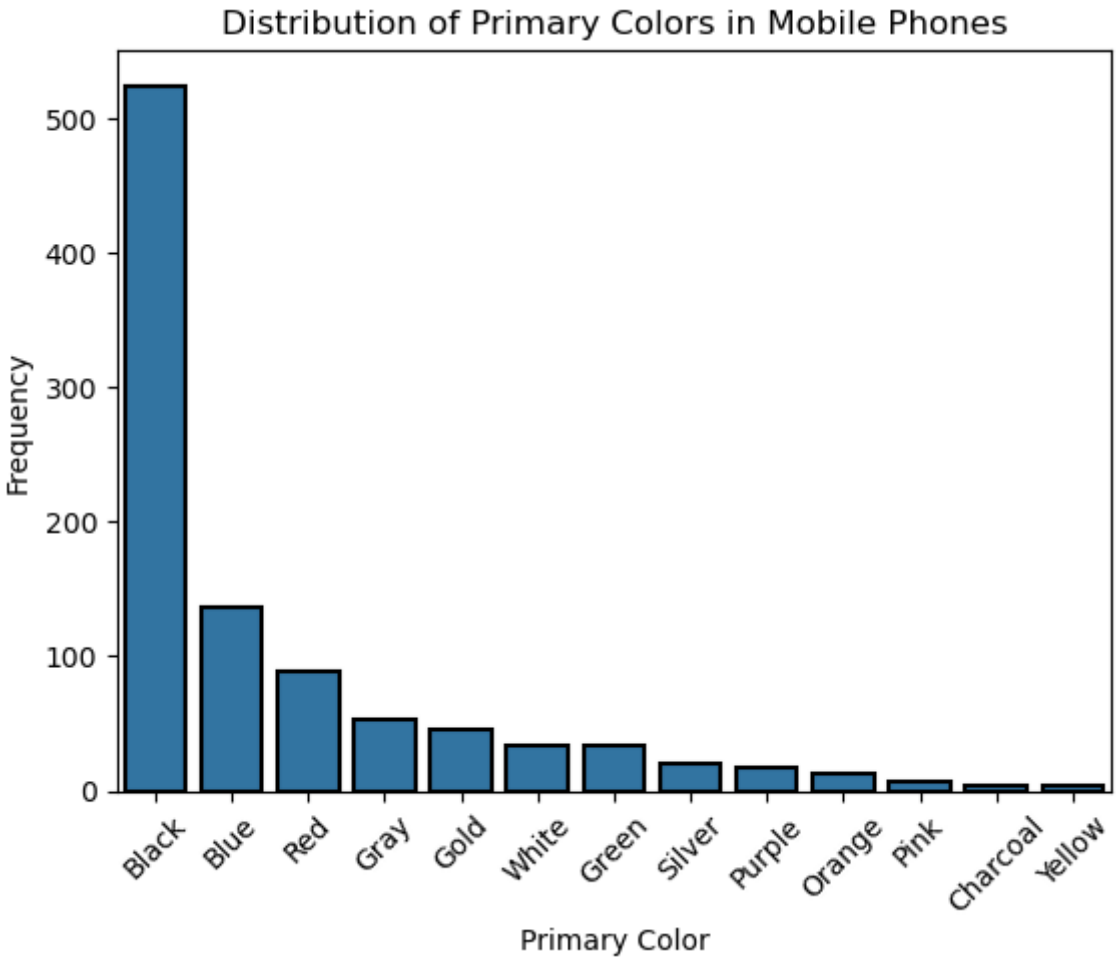
```
sellers amount mean is: 17.8876404494382 and the median is: 9.0
```

In [14]:
```
sns.histplot(df['best_price'].dropna())
plt.title('Distribution of best price')
plt.show()

print('best price mean is:',df['best_price'].mean(), "and the median is:", df['bes
```

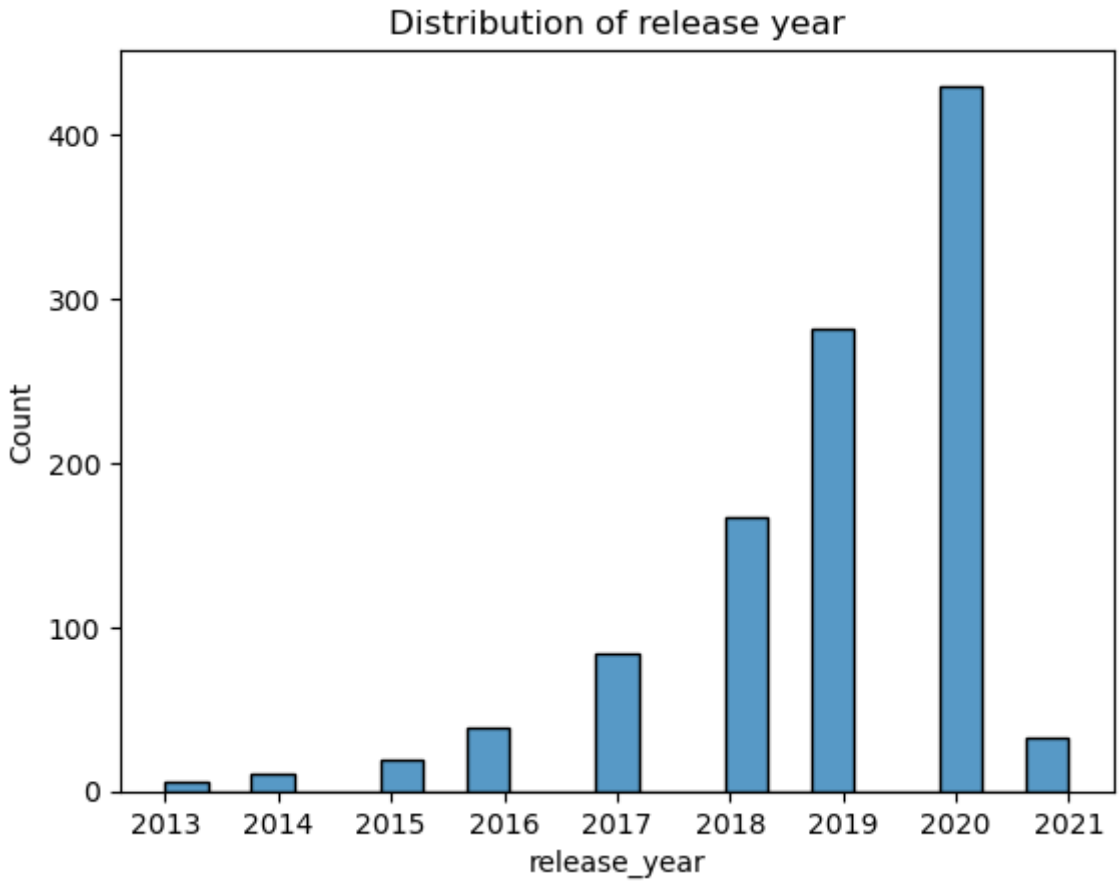



```
best price mean is: 8319.705992509364 and the median is: 4954.0
```

In [15]:
```
color_distribution = sns.countplot(data=df, x='primary_color', palette=[sns.color_
plt.title('Distribution of Primary Colors in Mobile Phones')
plt.xlabel('Primary Color')
plt.ylabel('Frequency')
plt.xticks(rotation=45)
plt.show()
```




In [16]:
```
df['release_date'] = pd.to_datetime(df['release_date'])
df['release_year'] = df['release_date'].dt.year
df[['release_date', 'release_year']].head()

sns.histplot(df['release_year'].dropna())
plt.title('Distribution of release year')
plt.show()
```




In [17]:
```
df['release_date'].max()
```

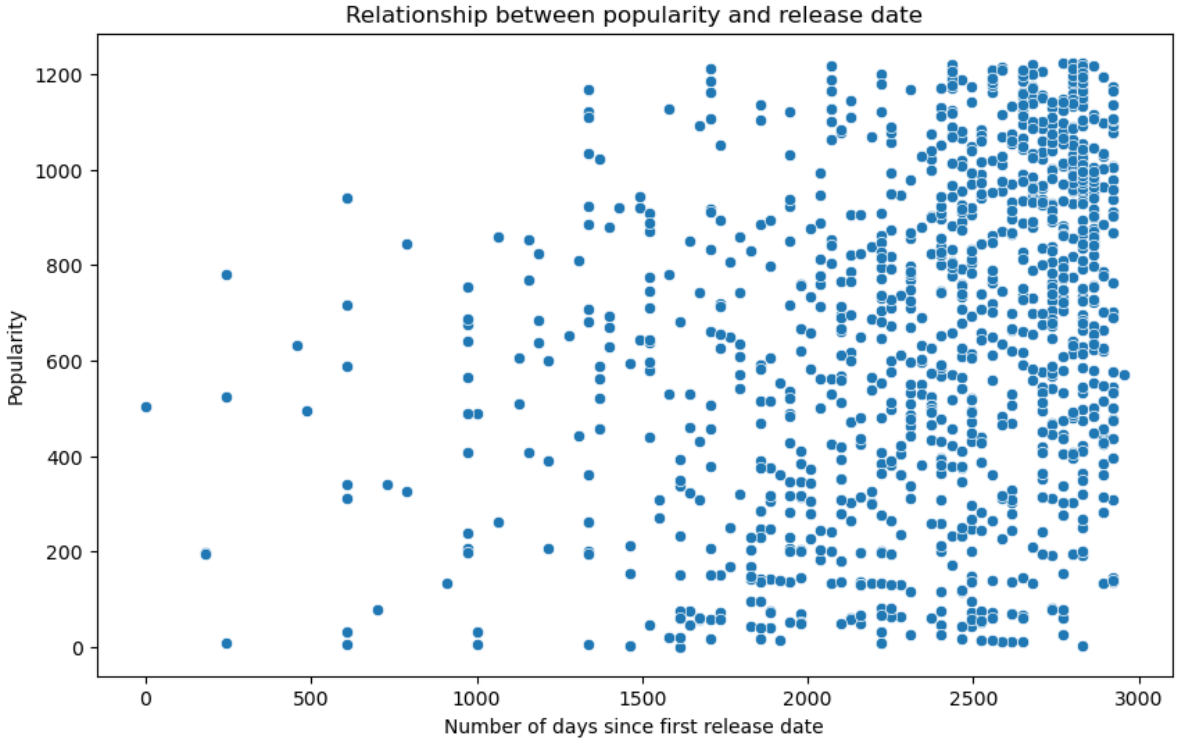
Out[17]:
```
Timestamp('2021-02-01 00:00:00')
```

המהגמה המורחבת היא שככל שהשנה גדולה יותר כך ישנם יותר דגמי טלפונים. עם זאת, נראה כי בשנת 2021 ישנו שינוי. את השינוי ניתן להסביר בכך שהמידע "קטום מימין". מהתבוננות במידע ניתן לראות כי האיסוף נגמר בחודש פברואר 2021, ולכן ישנם פחות דגמים משנה זו בנתונים. נשים לב לעניין נוסף, רוב המכשירים יצאו הבשנים מאוחרות יותר, מה שגורם לחוסר איזון בנתונים

## 2) קורולציה בין נתונים

בסעיף הקודם אמרנו שיכולה להיות בעייתיות כאשר נבחן את הפופלריות בתלות בשנת יצאית המכשיר. נבחן מקרה זה ונראה אם קיים קשר ואיך נצתרך לבחון אותו בראי הבעיתיות שציינו. נציין כי אנחנו מצפים לראות קשר חזק בין שנת יציאת המכשיר לבין הפופלריות שלו.

In [18]:
```
# Converting the release_date to release_date - days from first release, so we can
df['release_date'] = pd.to_datetime(df['release_date'], errors='coerce')
df['release_days'] = (df['release_date'] - df['release_date'].min()).dt.days

plt.figure(figsize=(10, 6))
sns.scatterplot(x='release_days', y='popularity', data=df)
plt.title('Relationship between popularity and release date')
plt.xlabel('Number of days since first release date')
plt.ylabel('Popularity')
plt.show()

correlation_popularity_release_date = df['popularity'].corr(df['release_days'])
correlation_popularity_release_date
```

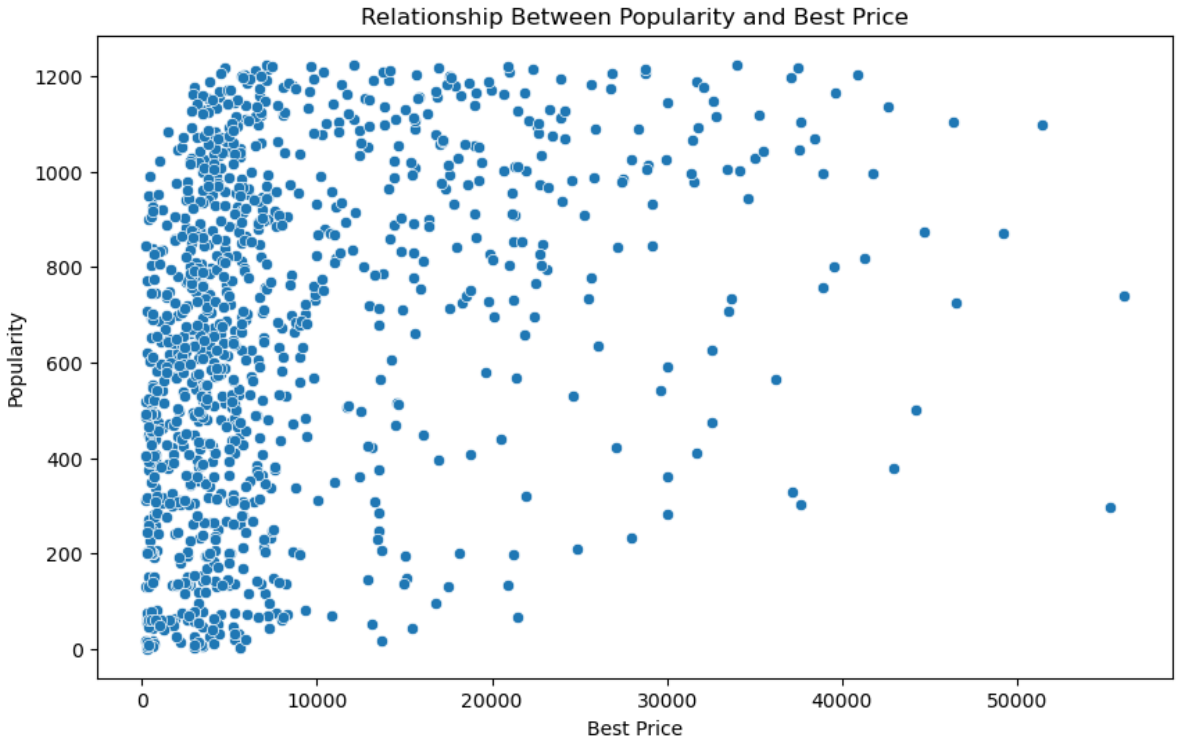



Out[18]: 0.3152615081116024

קיבלנו כי קיימת קורולציה חיובית, עם זאת, קורולציה של 0.31 היא לא מאוד חזקה. התוצאה היא אכן דומה למה שציפינו על אף שציפינו לקורולציה חזקה יותר. דבר זה יכול להצביע על כך שישנם יותר גורמים התורמים לפופולריות של מכשיר מאשר ה"חדשנות" שלו.

נבחן את הקשר בין המחיר "הטוב ביותר" של המכשיר לבין הפופלריות שלו. ראינו כי רוב המכשירים הם בטווח המחירים הנמוכים יותר, לכן אנחנו מצפים כי מכשירים אלו יהיו פופלרים יותר - ושנראה קורולציה שלילית.

In [19]:
```
plt.figure(figsize=(10, 6))
sns.scatterplot(x='best_price', y='popularity', data=df)
plt.title('Relationship Between Popularity and Best Price')
plt.xlabel('Best Price')
plt.ylabel('Popularity')
plt.show()

correlation_popularity_best_price = df['popularity'].corr(df['best_price'])
correlation_popularity_best_price
```

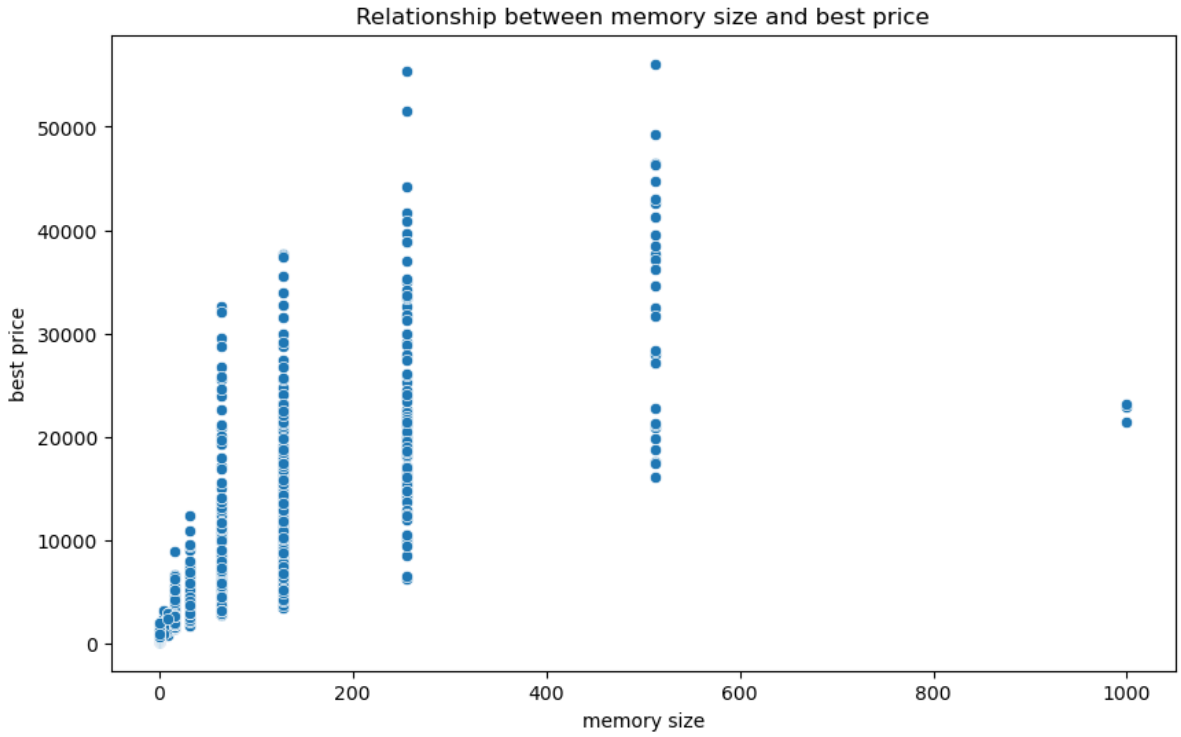



Out[19]: 0.34416605746513085

התוצאות שקיבלנו הפתיעו אותנו. קיבלנו קורלציה חיובות מה שאומר שככל שהמכשיר יקר יותר כך הוא יותר פופלרי, על אף שהקורולציה היא סביב 0.34 ולא מאוד חזקה.

בעקבות הפתעה זו נרצה לבחון האם יש גורמים אחרים המשפיעים על המחיר, נבדוק לדוגמא את הקורלציה בין זיכרון למחיר

In [20]:
```
plt.figure(figsize=(10, 6))
sns.scatterplot(x='memory_size', y='best_price', data=df)
plt.title('Relationship between memory size and best price')
plt.xlabel('memory size')
plt.ylabel('best price')
plt.show()

correlation_popularity_release_date = df['memory_size'].corr(df['best_price'])
correlation_popularity_release_date
```




Out[20]: 0.7115133191866106

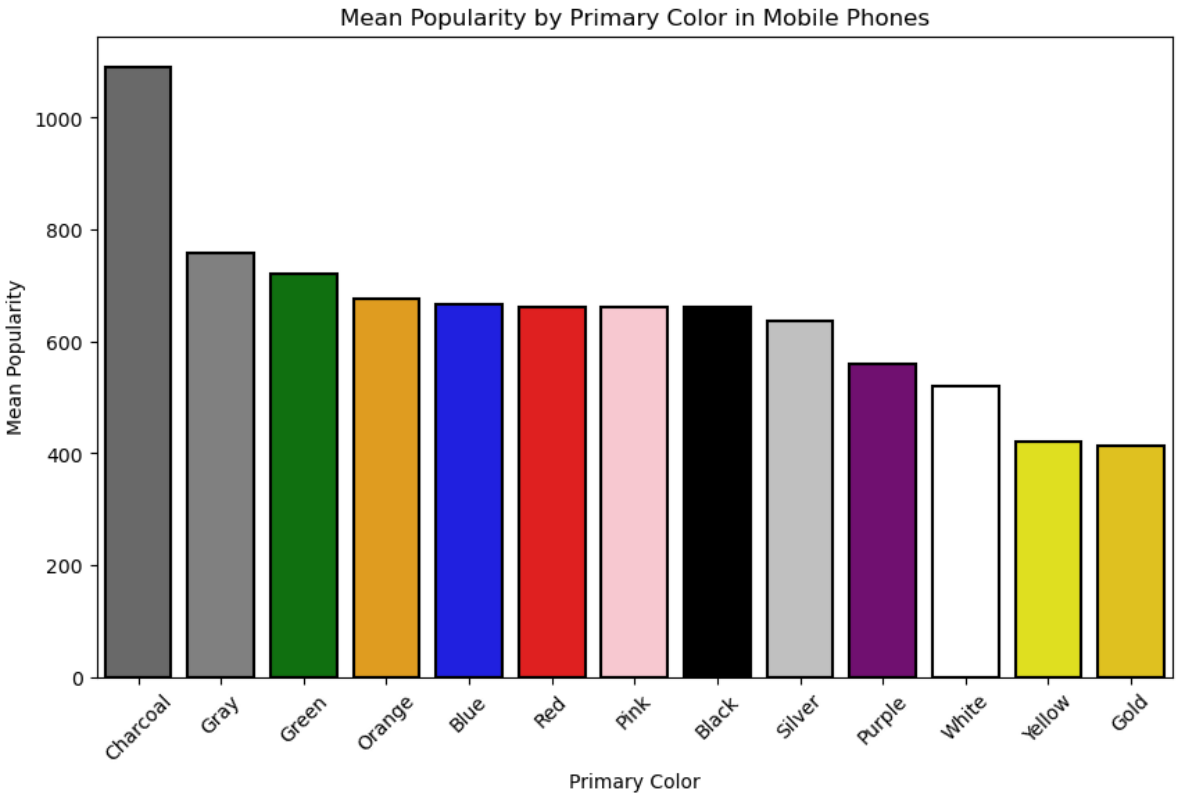
ביצענו בדיקה על מנת לראות האם יתכן כי לקוחות קונים פלאפונים יקרים בגלל שהם טובים יותר (למשל, זיכרון גדול)

על אף החוסרים בערכי הזיכרון, בחרנו להציג קורלציה זו ומכיוון שקיבלנו קורלציה חזקה דיו, נוכל להסיק כי על אף החוסרים, קיימת קורלציה בין גודל הזיכרון למחיר ולכן לקוחות שירצו פלאפונים עם זכרון גדול יותר ייאלצו לשלם יותר.

נרצה לבחון את הקשר בין צבע המכשיר לבין הפופולריות שלו. נשים לב כי חסרים כ90 רשומות, אין דרך להשלים את הרשמות בצורה נכונה, לכן נבחר להתעלם מהמכשירים ללא הצבע. משום שהחוסרים הם פחות מ10% נוכל להשתמש במסקנות מהבדיקה, אך עלינו לקחת את חוסרים אלו בחשבון. נבחן תחילה את הפופולריות הממוצעת של כל צבע

In [21]:
```
color_popularity_summary = df.groupby('primary_color')['popularity'].mean().reset_
color_popularity_summary = color_popularity_summary.sort_values(by='popularity', a
color_model_count = df['primary_color'].value_counts().reindex(color_popularity_su

# give boxes the same color that they represent
color_map = {
        'Black': 'black',
        'White': 'white',
        'Blue': 'blue',
        'Red': 'red',
        'Green': 'green',
        'Yellow': 'yellow',
        'Orange': 'orange',
        'Purple': 'purple',
        'Pink': 'pink',
        'Gray': 'gray',
        'Silver': 'silver',
        'Gold': 'gold',
        'Charcoal': 'dimgray'
}

plt.figure(figsize=(10, 6))
```

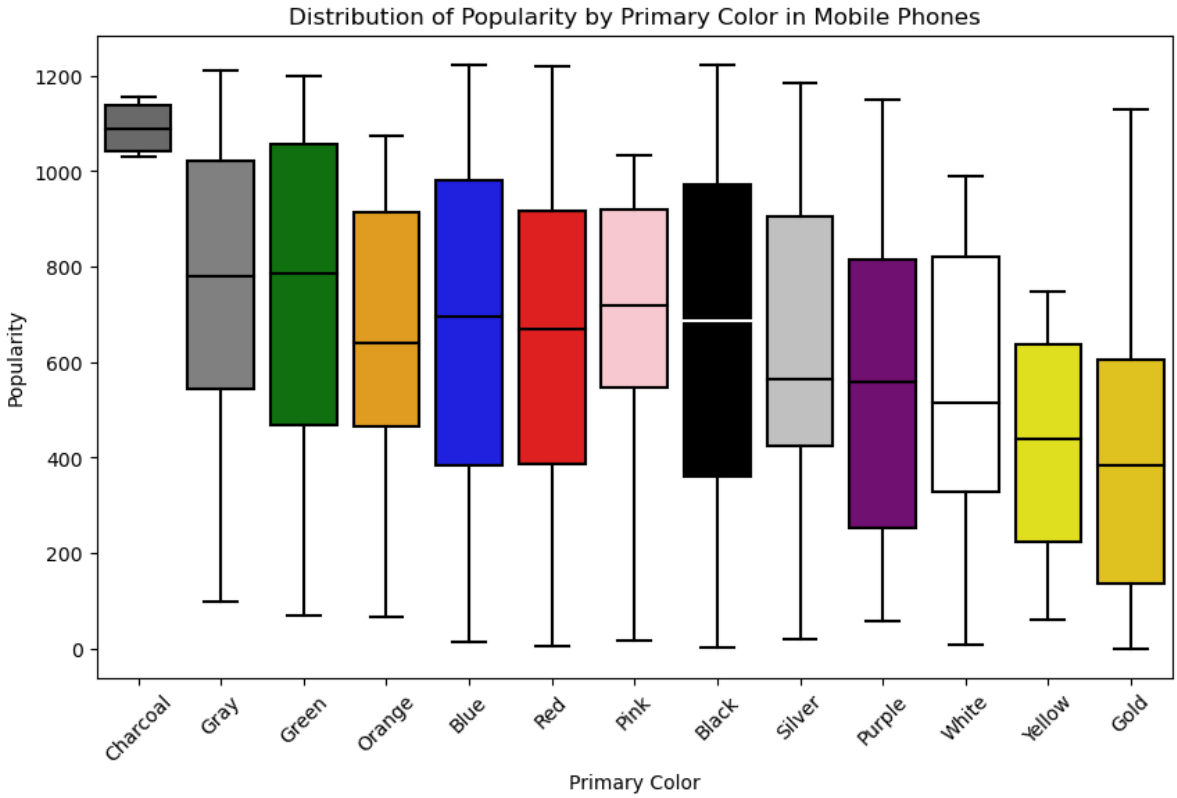
```
sns.barplot(data=color_popularity_summary, x='primary_color', y='popularity', pale
plt.title('Mean Popularity by Primary Color in Mobile Phones')
plt.xlabel('Primary Color')
plt.ylabel('Mean Popularity')
plt.xticks(rotation=45)
plt.show()
```




ניזכר בגרף שציירנו מוקדם יותר שמייצג את כמות המכשירים בכל אחד מהצבעים, שם ראינו כי יש חוסר איזון בכמות המכשירים שיש בכל אחד מהצבעים. נשתמש בגרף קופסאות על מנת לקבל ראיה יותר מדוייקת של הפופלריות של כל אחד מהצבעים

In [22]:

```
plt.figure(figsize=(10, 6))
box_plot = sns.boxplot(data=df, x='primary_color', y='popularity', palette=color_ma
plt.title('Distribution of Popularity by Primary Color in Mobile Phones')
plt.xlabel('Primary Color')
plt.ylabel('Popularity')
plt.xticks(rotation=45)

# make black median line white
black_index = color_popularity_summary['primary_color'].tolist().index('Black')
ax = plt.gca()
black_median_line = ax.lines[black_index * 6 + 4]
black_median_line.set_color('white')

plt.show()
```




מהגרף ראינו שמכשירים בצבע פחם כולם פופולרים מאוד, אך חשוב לזכור שאנחנו יודעים שיש מעט מעוד מהם בנתונים שלנו. כמו כן אנחנו רואים שיש מכשירים פופלארים מאוד בכמעט כל צבע, אם זאת לדוגמא, לרוב מכשיר ירוק יהיה פופולרי יותר ממכשיר צהוב וזאת לפי החציון הפופולרי של כל אחד מהם. בנוסף, מכשירים זהובים וצהובים לרוב הכי פחות פופולרים, על אף שישנה שונות מאוד גדולה במכשירים הזהובים.

# part 3

## 1) הגדרת השערה

מניסיונו האישי - "דומיין נולג'" מכשירים גדולים יותר הם לרוב פופולרים יותר. נרצה לבחון אם השארה זו נכונה. לשם כך, נגדיר מכשיר גדול כמשיר עם מסך מעל 5 אינץ'.

## 2) הגדרת ההשערות

H0 - הפופלריות הממצועת בין מכשירים עם מסך גדול (מעל 5 אינץ') לבין מכשירים עם מסך קטן (מתחת ל5 אינץ') היא זהה

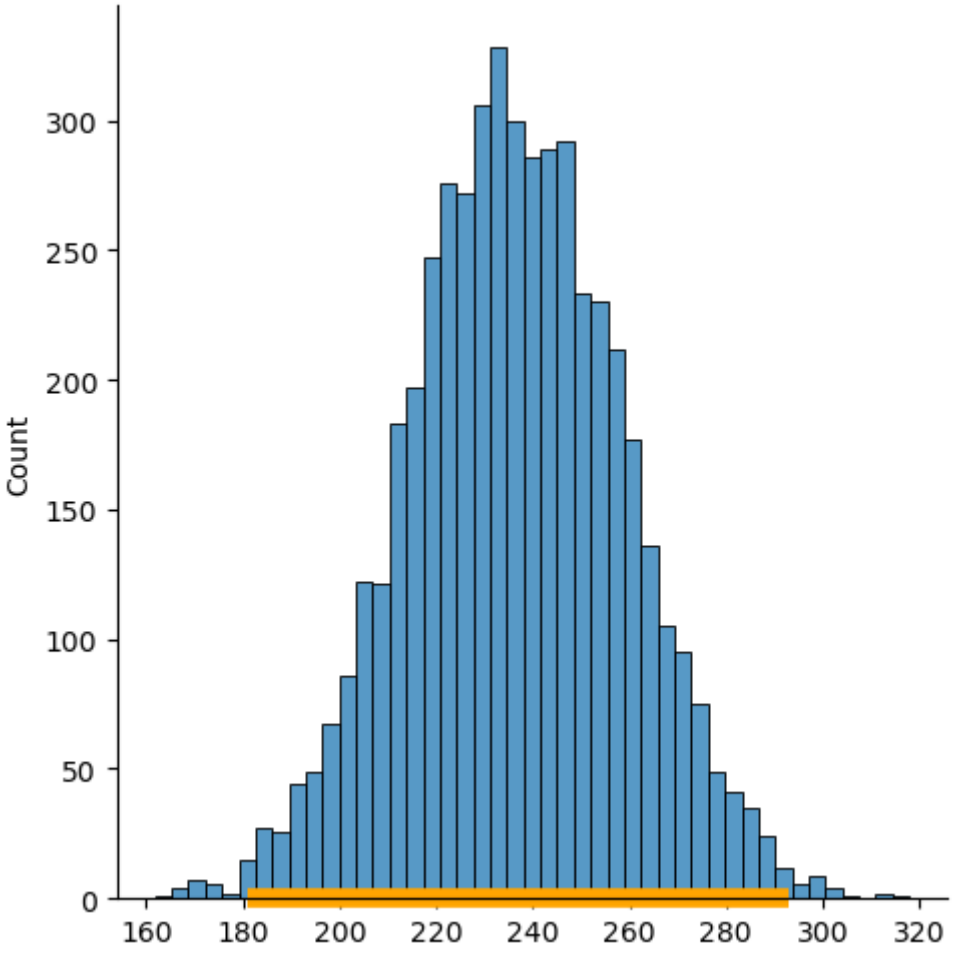
H1 - הפופלריות הממצועת בין מכשירים עם מסך גדול (מעל 5 אינץ') לבין מכשירים עם מסך קטן (מתחת ל5 אינץ') היא שונה

## 3) בחינת השערות

In [23]:
```
# Creating the new columns based on screen size
df['is_big_phone'] = np.where(df['screen_size'] > 5, 'yes', 'no')

# function that returns the difference in popularity averages
def diff_of_avgs(df, column_name, grouping_var):
    grpby_var = df.groupby(grouping_var)
```

```
    avgs = grpby_var[column_name].mean()
    return avgs.loc['yes'] - avgs.loc['no']


def bootstrap_mean_difference(original_sample, column_name, grouping_var, num_repl
    original_sample_size = original_sample.shape[0]
    original_sample_cols_of_interest = original_sample[[column_name, grouping_var]
    bstrap_mean_diffs = np.empty(num_replications)
    for i in range(num_replications):
        bootstrap_sample = original_sample_cols_of_interest.sample(original_sample_
        resampled_mean_diff = diff_of_avgs(bootstrap_sample, column_name, grouping_
        bstrap_mean_diffs[i] = resampled_mean_diff
    return bstrap_mean_diffs

# run the bootstrap procedure
bstrap_diffs = bootstrap_mean_difference(df, 'popularity', 'is_big_phone',  5000)

# Get the endpoints of the 95% confidence interval
left_end = np.percentile(bstrap_diffs, 0.5, method='higher')
right_end = np.percentile(bstrap_diffs, 99.5, method='higher')
print('The 99% boostsrap confidence interval for difference between population mear

# visualize results
ax = sns.displot(bstrap_diffs)
plt.hlines(y=0, xmin=left_end, xmax=right_end, colors='orange', linestyles='solid'
```

```
The 99% boostsrap confidence interval for difference between population means [18
0.7956160777564, 292.90402430854624]
```




ניתן לראות כי 0 לא נמצא ברווח הסמך, לכן נדחה את השערת ה0 - שאומרת כי הפופולריות של מכשירים גדולים ומכשירים קטנים היא זהה. תוצאת מבחן בדיקת ההשערות מחזקת את הטענה שהעלנו לפני המבחן. אם זאת, לא ניתן להגיע למסקנה חד משמעית עם מבחן מסוג זה, משום שלא בוצע נסיוי רשמי.

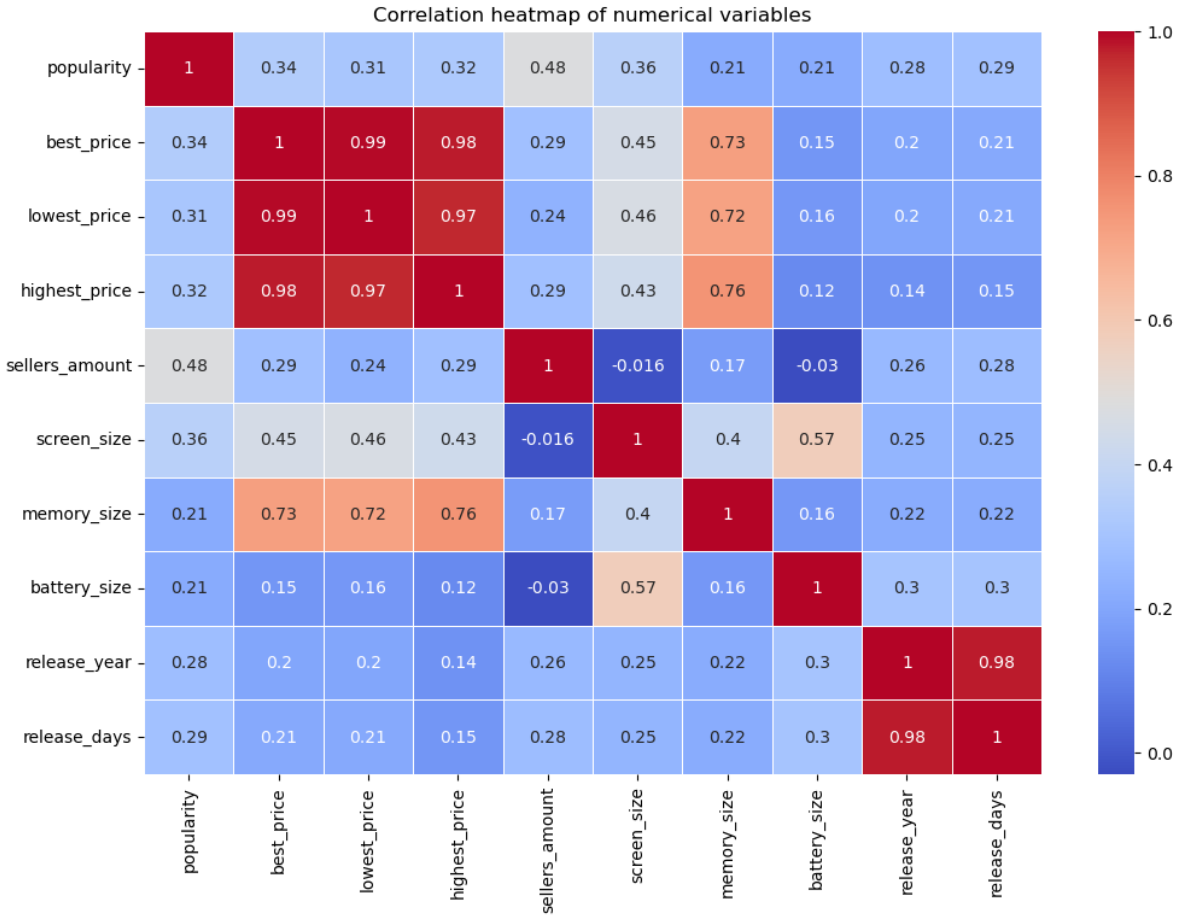
# part 4

נרצה לאמן מסווג שבהינתן מספר משתנים שונים יוכל לשערך את הפופולריות של מכשיר חדש שיכנס לשוק. ננסה לסווג מכשיר חדש ולראות אם הוא יהיה בחציון הראשון של המכשירים הפופולרים בשוק. נחפש את המשתנים הטובים היותר. נשתמש ב"טבלת חום" עבור המשתנים הנומרים. לא נשתמש במשתנה של תאריך יציאת המכשיר כי החברה לא שולטת בתאריך יציאת המכשיר - הוא תמיד יהיה החדש ביותר כשהוא יצא.

In [24]:
```
from sklearn.model_selection import train_test_split
# split to train and test set

shuffled_df = df.sample(frac=1, random_state=42)
df_train, df_test = train_test_split(shuffled_df, test_size=0.2)
```

In [25]:
```
import warnings
correlation_matrix = df_train.corr()

warnings.filterwarnings("ignore")
plt.figure(figsize=(12, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', linewidths=0.5)
plt.title('Correlation heatmap of numerical variables')
plt.show()
```

```
C:\Users\boazb\AppData\Local\Temp\ipykernel_12856\3178363212.py:2: FutureWarning:
The default value of numeric_only in DataFrame.corr is deprecated. In a future ver
sion, it will default to False. Select only valid columns or specify the value of
numeric_only to silence this warning.
  correlation_matrix = df_train.corr()
```



נשים לב כי הקורולציות בין המחיר הטוב ביותר לבין המחיר הגבוה והמחיר הנמוך מאוד גבוהות - כמעט 1. משום שקיימים נתונים חסרים רבים גם במחיר הגבוה וגם במחיר הנמוך, נבחר לעבוד רק עם המחיר הטוב יותר.

נבחר לאמן את המסווג על כל שאר המשתנים, למעט גודל הזיכרון. נבחר שלא להשתמש בגודל הזיכרון משום שהוא בעל הקורולציה הנמוכה ביותר מכל המשתנים וגם יש לו חוסרים רבים. סיבה נוספת לוותר אליו כי אנחנו משתמש ב4 משתנים שונים ונרצה להימנע מקללת המימדים משום שיש לנו רק 1068 רשומות

נשים לב כי הקורולציות בין המשתנים שבחרנו לבין משתנה ה"פופלריות" לא מאוד חזקות, ולכן נצפה כי המסווג לא יהיה חזק במיוחד. אם זאת, אנחנו חושבים שהמסווג הזה הוא חשוב מאוד למרות הבעיות שאחנו צופים, כי חברות הטלפונים מעוניינות בסוף בפופולריות של המכשיר וזה המניע המרכזי שלהן כאשר הן מוציאות מכשיר חדש לשוק ועובדות על הפיתוח שלו. לכן גם מסווג לא מושלם יהיה שימושי מאוד.

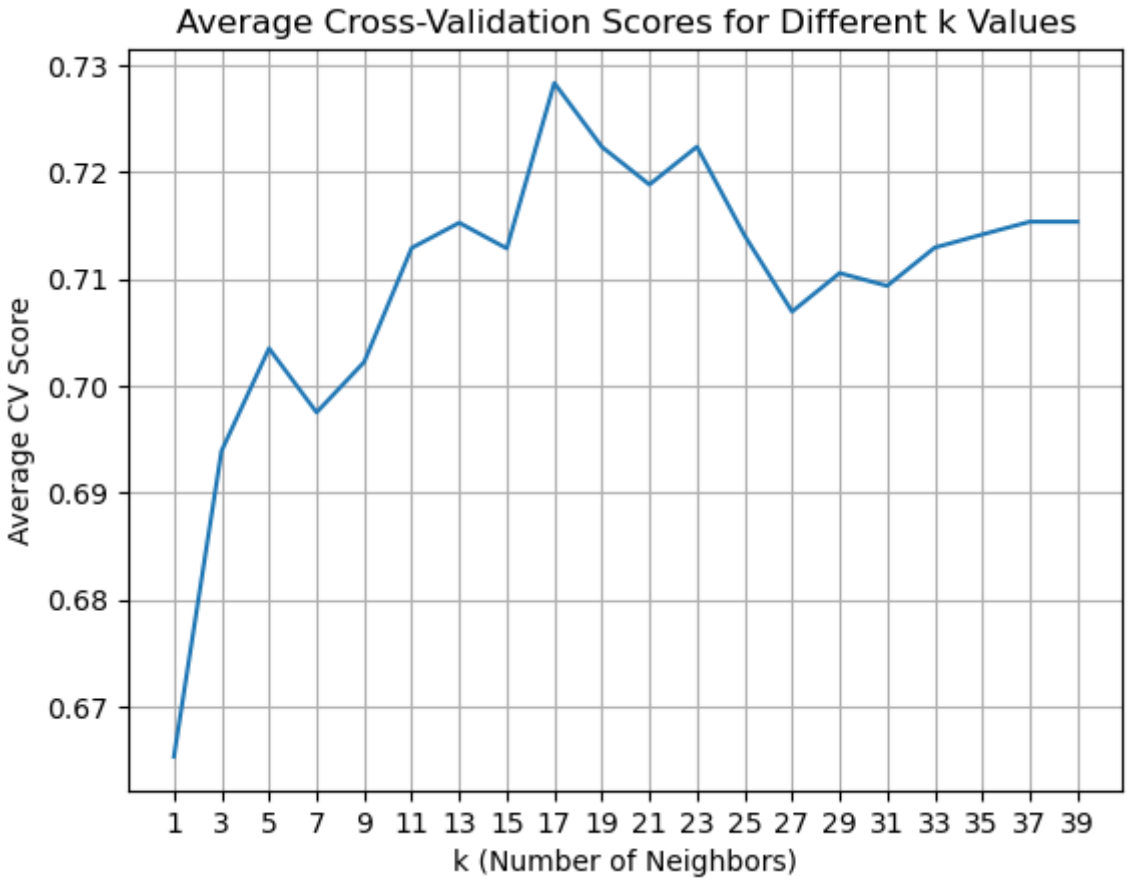
In [26]:
```
# preproces
# 1. Remove rows with missing values in the columns of interest
df_train_clean = df_train.dropna(subset=["best_price", "sellers_amount", "screen_s
df_test_clean = df_test.dropna(subset=["best_price", "sellers_amount", "screen_size


# 2. Define rows with popularity values between 1 and 150 as "popular" and the rest
df_train_clean['popularity_label'] = df_train_clean['popularity'].apply(lambda x: 1
df_test_clean['popularity_label'] = df_test_clean['popularity'].apply(lambda x: 1


# 3. remove un relevent columns from df and shuffel
features = ["best_price", "sellers_amount", "screen_size", "battery_size", "popular
relevant_df_train = df_train_clean.loc[:, features]
relevant_df_test = df_test_clean.loc[:, features]
```

In [27]:
```
# split to test and train

# define which variables are the features and which are the labels
X_train = relevant_df_train[["best_price", "sellers_amount", "screen_size", "batter
Y_train = relevant_df_train["popularity_label"]  # labels

X_test = relevant_df_test[["best_price", "sellers_amount", "screen_size", "battery_
Y_test = relevant_df_test["popularity_label"]  # labels
```

In [28]:
```
from sklearn.preprocessing import StandardScaler
# scaling

df_columns = X_train.columns
scaler = StandardScaler()
scaled_X_train = scaler.fit_transform(X_train)
scaled_X_test = scaler.transform(X_test)
scaled_df = pd.DataFrame(scaled_X_train, columns=df_columns)
```

In [29]:
```
from sklearn.model_selection import cross_val_score
from sklearn.preprocessing import MinMaxScaler
from sklearn.neighbors import KNeighborsClassifier
# CV

k_avg_score = np.zeros(40)
for k in range(1, 40, 2): # take only odd
    knn_model = KNeighborsClassifier(n_neighbors=k)
```

```
    cv_scores = cross_val_score(knn_model, scaled_X_train, Y_train, cv=10)
    k_avg_score[k] = cv_scores.mean()

best_k = np.argmax(k_avg_score)

import matplotlib.pyplot as plt

# Plot the average scores for odd k values
plt.plot(range(1, 40, 2), k_avg_score[1:40:2])
plt.title('Average Cross-Validation Scores for Different k Values')
plt.xlabel('k (Number of Neighbors)')
plt.ylabel('Average CV Score')
plt.xticks(range(1, 40, 2))  # Set x-axis ticks to show only odd k values
plt.grid(True)
plt.show()

print('Highest accuracy is obtained for k =', best_k, 'and equals', max(k_avg_scor
```




```
Highest accuracy is obtained for k = 17 and equals 0.7282913165266106
```

In [30]:

```
#train a classifier

knn_classifier = KNeighborsClassifier(n_neighbors= best_k)
knn_classifier.fit(scaled_X_train, Y_train)
print('accuracy of the classifier is', knn_classifier.score(scaled_X_test, Y_test)
```

```
accuracy of the classifier is 0.7370892018779343
```

In [31]:

```
from sklearn.metrics import confusion_matrix, precision_score, recall_score
# Compute a confusion matrix

predictions = knn_classifier.predict(X=scaled_X_test)
# (0,0) = true negatives, (1,0) = false negatives, (1,1) = true positives, (0,1) =
print('confusion matrix: \n', confusion_matrix(y_true=Y_test, y_pred=predictions))
```

```
print('precision: ', precision_score(y_true=Y_test, y_pred=predictions, pos_label=1
print('recall: ', recall_score(y_true=Y_test, y_pred=predictions, pos_label=1))
```

```
confusion matrix:
 [[69 25]
 [31 88]]
precision:  0.7787610619469026
recall:  0.7394957983193278
```

קיבלנו מסווג עם רמת דיוק של כ74%. כמו כן, הפרסז'ן הוא 78%, והריקול הוא 74%. באופן כללי, המסווג דיי טוב ומאוזן בסיווג של מכשירים פופולאריים או לא פופולאריים, אך עם זאת הוא עדיין בעל רמת דיוק של כ74% שזה לא מושלם. היינו ממליצים לחברות להשתמש במסווג הנ"ל אך להשתמש בו רק על מנת לקבל אינדיקציה ראשונית ולא לקבל החלטות סופיות באמצעותו.

על מנת לשפר את ביצועי המסווג היינו עושים את הדברים הבאים:

1. אוספים נתונים נוספים, היינו רוצים להרחיב את מסד הנתונים עד לשנת 2023. שינוי זה היה מגיד את מספר הנתונים שלנו ולכן המסווג היה מדויק יותר וגם היינו משקפים מגמות חדשניות יותר בשוק.
2. היינו רוצים להשלים את החוסרים עבור המשתנה של מערכות ההפעלה ואז היינו יכולים להוסיף אותו למסווג באמצעות שיטת "קידוד-אחד-חם".
3. במידה והייתה לנו גישה למדינות נוספות, או לפחות למדינה אחת מכל יבשת או אזור בעל מאפיינים תרבותיים דומים היינו יכולים להציג תמונת מצב בינלאומית שלא משקפת רק את המגמות באוקרינה.


מיני פרויקט

מבוא לניתוח נתונים

בועז בלומו 209120195

בר גמליאל 318651734

נעשה שימוש בGPT – השיחות מצורפות בהגשה

## שאלות

השאלה המרכזית שליוותה אותנו במהלך הפרויקט היא "איך מאפיינים שונים משפיעים על הפופולאריות של מכשיר סלולארי?"

שאלה זו עניינה אותנו במיוחד מכיוון שמטרתה של חברה לייצור פלאפונים היא להוציא מכשירים כמה שיותר פופולריים ועליהם לחזות זאת כבר בשלב התכנון של המכשיר.
במידה והחברה תתכנן מכשיר בעל מאפיינים מסוימים היא תוכל להעריך את פופולאריות המכשיר וכך תקבל אינדיקציה לגבי כדאיות הפיתוח שלו.

תחת השאלה המרכזית יש שאלות נוספות:

1. האם פלאפונים חדשניים הם פופולאריים יותר?
2. האם גודל המסך של מכשיר משפיע על רמת הפופולאריות שלו?
3. האם ניתן לחזות פופולאריות של מכשיר על סמך מאפייניו?
4. כיצד משפיע צבע המכשיר על הפופולריות שלו?

בנוסף, בנימה אישית, בזמן העבודה על הפרויקט אחד מאיתנו היה בתהליך של חיפוש מכשיר שיחליף את המכשיר הישן שלו ועניין אותנו לדעת איך מכשירים שונים נהים פופולאריים בראי הקניה של המכשיר

Data Set

השתמשנו במסד הנתונים phones_data המכיל מידע על טלפונים שהיו זמינים לרכישה באוקראינה בשנים האחרונות.
מסד זה מכיל את הפיצ'רים הבאים:

שם היצרן
שם הדגם
מערכת ההפעלה
פופולאריות (דירוג המכשירים מהפופולארי ביותר להכי פחות פופולארי)
המחיר הטוב ביותר (ככל הנראה בשקלול גורמים נוספים כמו שירות ואחריות (בUAH))
המחיר הנמוך ביותר (בUAH)
המחיר הגבוה ביותר (בUAH)
כמות המוכרים המשווקים את המכשיר
גודל המסך (באינצ'ים)
גודל הזיכרון (בGB)
גודל הסוללה (בmAh)
תאריך יציאת המכשיר
צבע המכשיר (ללא תת גוון)

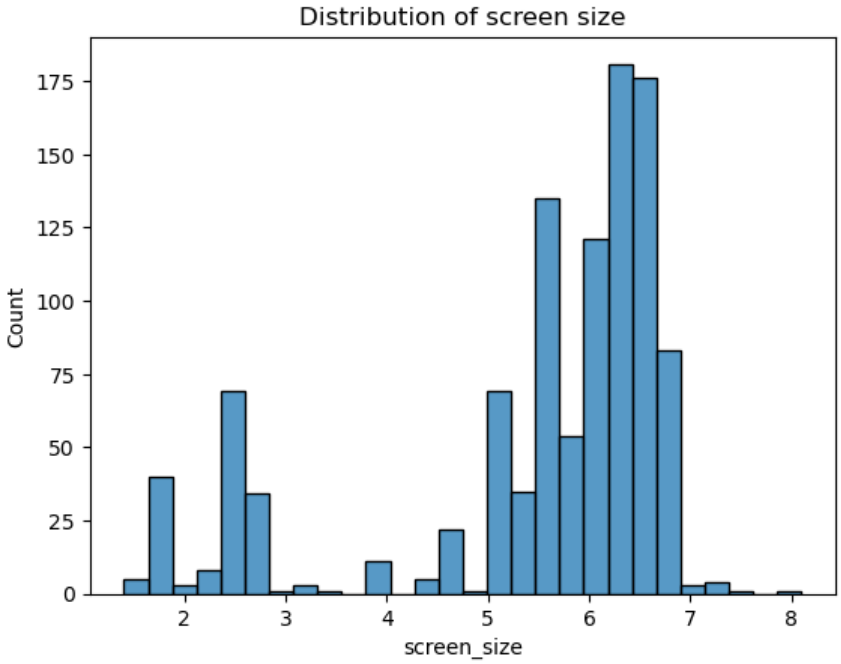
המכשירים במסד יצאו בין השנים 2013-2021
ישנם 1224 מכשירים המתוארים במסד, אך עם זאת, לאחר בדיקה גילינו כי קיימות 156 רשומות זהות לחלוטין אשר השוני היחיד ביניהן הוא הפופולאריות.
ההנחה שלנו היא כי הכפילויות נובעות מהטיית מדידה והשוני בפופולאריות נובע מהצורה בה הושמו ערכי הפופולריות למכשירים במסד, לכן בחרנו להסיר רשומות אלו ולהזין את ערך הפופולריות הממוצע (מבין אותן רשומות) לרשומה שנשארה.
כעת, במסד החדש, ישנן 1068 רשומות.
בנוסף, חסר מידע בחלק מהמכשירים.

מסד זה יעזור לנו לענות על שאלות המחקר שלנו מכיוון שהוא מכיל את פופולאריות המכשירים ובנוסף לכך, מכיל מידע על מאפייני המכשירים מה שיעזור לנו באמצעות ניתוח הנתונים להסיק מסקנות על הקשרים בין המאפיינים לפופולאריות המכשיר.

## ניתוח וממצאים

במהלך ניתוח הנתונים, התחלנו מהשאלות הכלליות ביותר על מנת להבין יותר טוב את המסד שלנו ולחוש את התנהגות הנתונים.
רצינו לראות איך מתפלגים מאפיינים שונים של המכשיר מתוך מחשבה שהדגמים הפופולאריים יהיו במרכז הצביר ולא בקצוות.
לדוגמא- התפלגות גודל המסך:

בהתפלגות זו ניתן לראות שמרבית המכשירים בעלי גודל מסך הנמצא בסביבות הגודל של 6 אינץ'

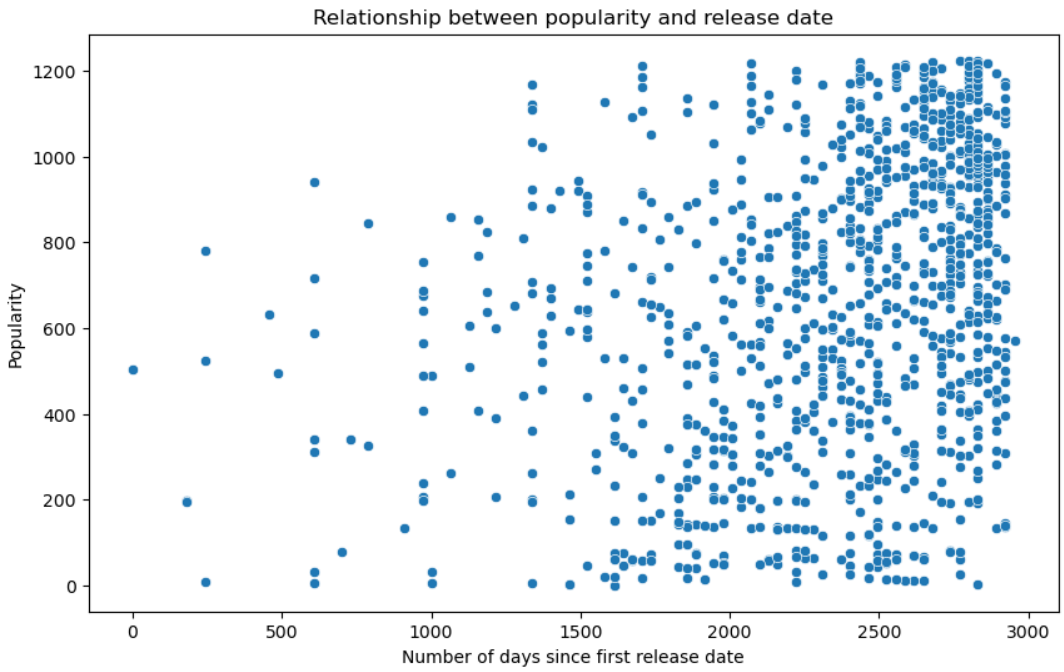



בנוסף, בדקנו התפלגויות של גודל סוללה, כמות מוכרים, המחיר הטוב ביותר וצבע המכשיר ובכולם הייתה התנהגות דומה בה רוב המכשירים התרכזו מסביב לערך מסוים.

בבדיקה שערכנו על התפלגות תאריכי היציאה של הפלאפונים במסד עלו מחשבות ומסקנות בנוגע להטיות הקיימות בנתונים עליהם נרחיב בחלק הרלוונטי.

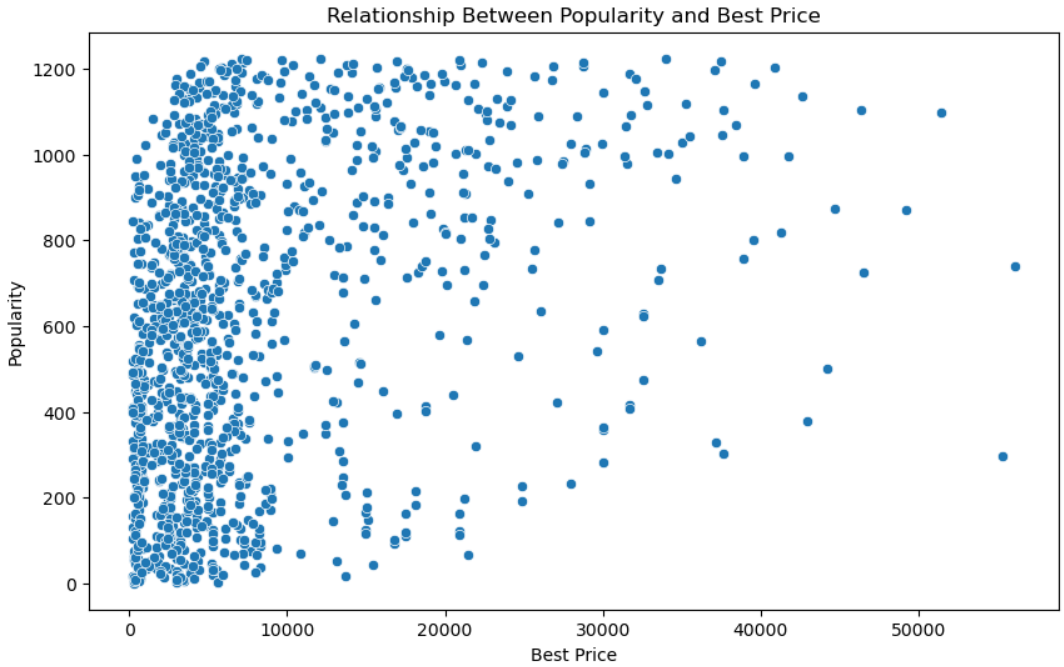
לאחר שקיבלנו תחושה על איך מאפיינים שונים מתנהגים רצינו להסתכל על קשרים בין מאפיינים לבין הפופולאריות של המכשירים.

חשבנו שיש קשר חזק בין תאריך היציאה של המכשיר לבין הפופולאריות שלו ורצינו לראות מה הקורלציה בין שני המשתנים האלו.




לאחר חישוב קיבלנו כי הקורלציה היא 0.31
אמנם קיבלנו כי קיימת קורלציה חיובית, אך עם זאת, קורלציה של 0.31 היא לא מאוד חזקה.
עפ"י התוצאה אכן קיימת קורלציה אך לא חזקה כפי שציפינו, מה שיכול להצביע על כך שישנם יותר גורמים התורמים לפופולריות של מכשיר מאשר ה"חדישות" שלו.

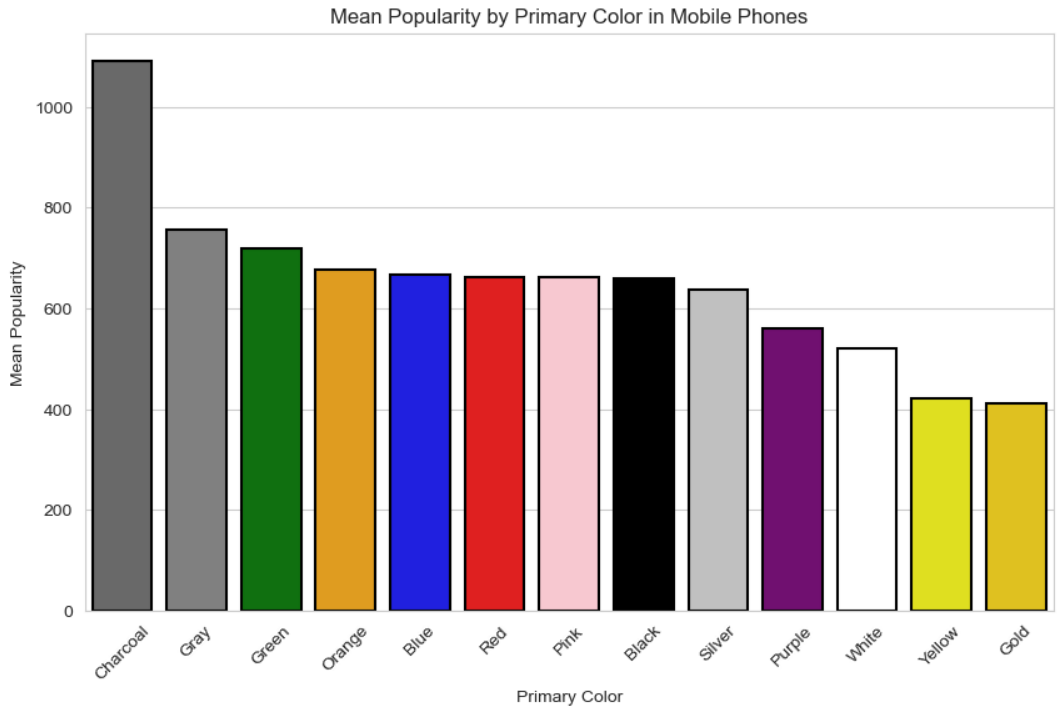
תופעה דומה גילינו כאשר חישבנו קורלציה בין פופולאריות למחיר כפי שניתן לראות בגרף:



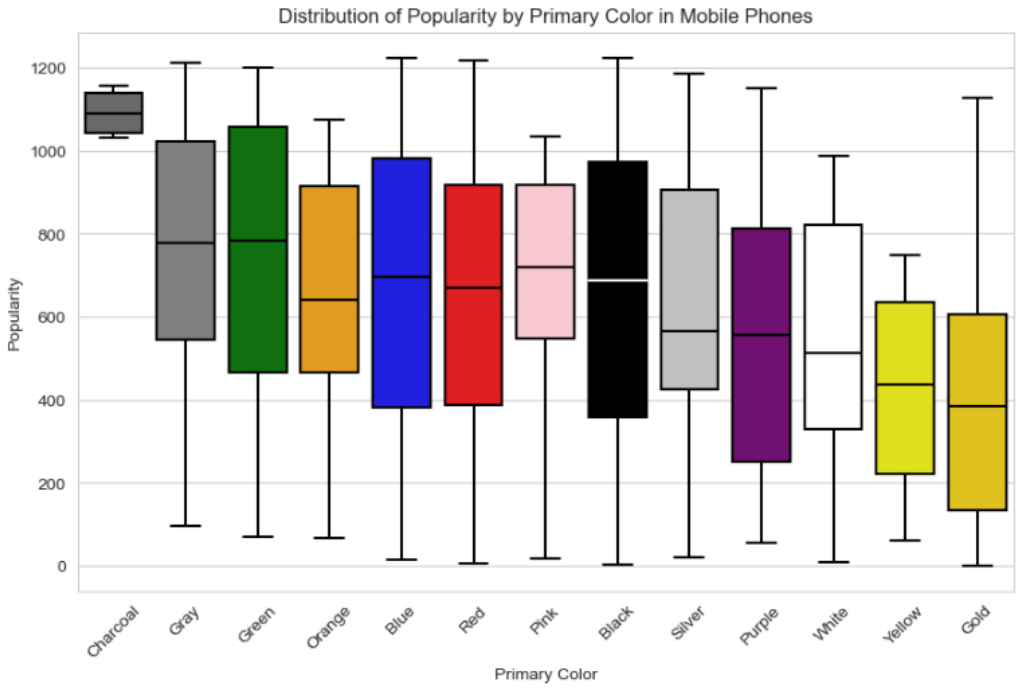

לאחר חישוב קיבלנו קורלציה של 0.34
נתון זה מעט מפתיע, מכיוון שהיינו מצפים שהמכשירים הזולים יותר יהיו יותר פופולאריים יותר (קורלציה שלילית).
יתכן כי דבר זה נובע מכך שהמכשירים היקרים יותר הם מתקדמים יותר והקדמה היא שגורמת

ליותר אנשים לקנות אותם על אף המחיר.
למשל, לאחר בדיקה, ראינו כי קיימת קורלציה חזקה (0.71) בין מחיר לבין גודל זיכרון וייתכן כי בגלל גודל הזיכרון לקוחות יקנו פלאפונים יקרים יותר.

רצינו לראות כיצד מתנהג משתנה הפופולריות ביחס לצבע המכשיר. לשם כך, בדקנו את הפופולריות הממוצעת של כל אחד מהצבעים:




לאחר שבדקנו זאת, רצינו לחקור את הנושא מעט יותר לעומק, במיוחד לאחר שראינו כי הצבע בו כמות המכשירים הגבוהה ביותר הוא הצבע עם הפופולריות הממוצעת הגבוהה ביותר. בחרנו להציג את הקשר באמצעות box plot, כך נוכל לראות גם את הטווח בין הערך המקסימלי למינימלי וגם היכן נמצאים רוב ציוני הפופולריות.

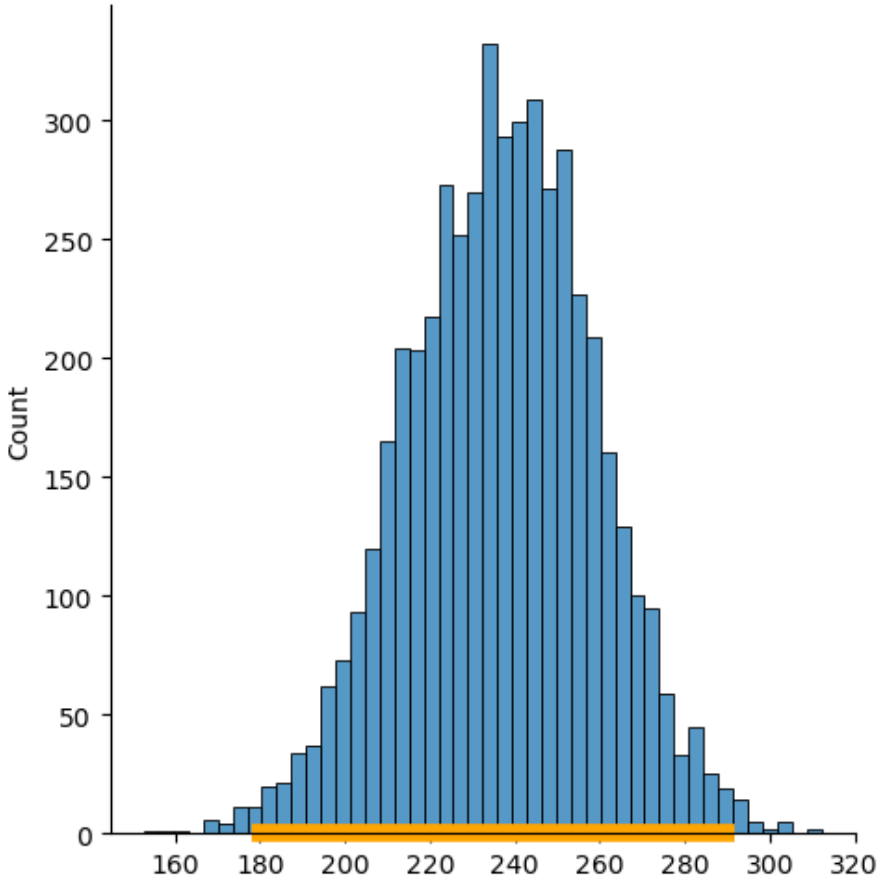


בעקבות גרפים אלו עלו לנו מספר מסקנות מעניינות.
ראינו שמכשירים בצבע פחם כולם פופולריים מאוד, אך עם זאת, פופולריות זו לא מעידה בהכרח על כך שהלקוח יעדיף לקנות מכשיר בצבע זה מכיוון שמספר הדגמים בצבע זה נמוך מאוד. כמו כן אנחנו רואים שיש מכשירים פופולאריים מאוד וגם לא פופולריים בכלל כמעט בכל צבע. עם זאת לדוגמא, לרוב מכשיר ירוק יהיה פופולרי יותר ממכשיר צהוב וזאת לפי החציון הפופולרי של כל אחד מהם.
בנוסף, מכשירים זהובים וצהובים הם לרוב הכי פחות פופולריים, אל אף שישנה שונות מאוד גדולה במכשירים הזהובים.

בתור לקוחות הקונים מכשירים סלולאריים אנו מרגישים כי גודל מסך מהווה פקטור משמעותי בשיקול של איזה מכשיר לקנות, לכן החלטנו לבדוק באמצעות בדיקת השערות האם גודל המסך משפיע על הפופולאריות של המכשיר.
בעקבות ההתפלגות שראינו מוקדם יותר החלטנו שמכשיר גדול יהיה מכשיר בעל מסך גדול מ5 אינץ'

השערת האפס- הפופולריות הממוצעת בין מכשירים עם מסך גדול (מעל 5 אינץ') לבין מכשירים עם מסך קטן (מתחת ל5 אינץ') היא זהה.
השערה אלטרנטיבית- הפופולריות הממוצעת בין מכשירים עם מסך גדול (מעל 5 אינץ') לבין מכשירים עם מסך קטן (מתחת ל5 אינץ') היא שונה.

בחרנו לבדוק את ההשערות באמצעות בוטסטראפ עם הסטטיסטי של הפרש ממוצע הפופולאריות בין פלאפונים מעל 5 אינץ' לבין פלאפונים מתחת ל5 אינץ'
קיבלנו את התוצאה הבאה:




ניתן לראות כי 0 לא מוכל ברווח הסמך ולכן נדחה את השערת האפס.

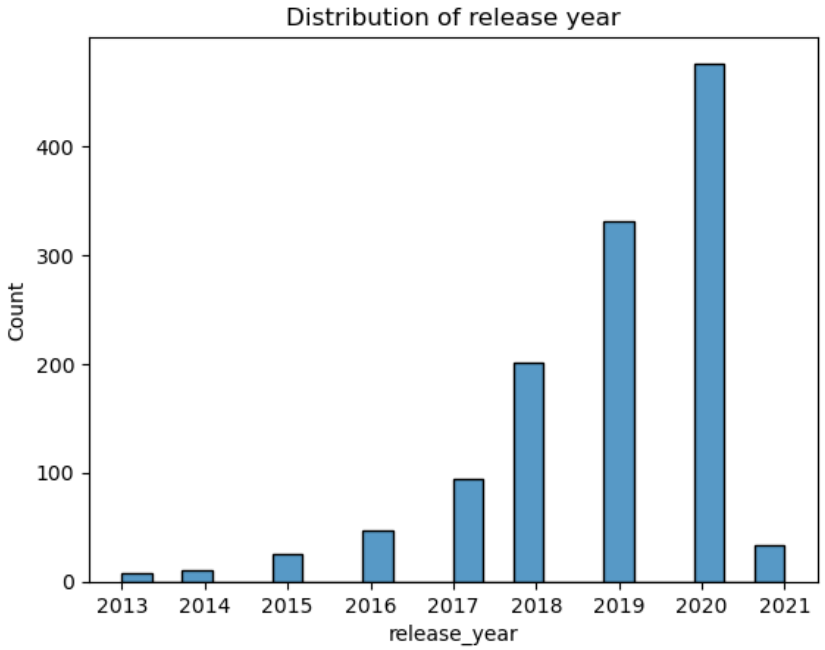
מבחן זה מעיד כי קיים הבדל בפופולאריות בין פלאפונים קטנים לגדולים.
מסקנה זו תואמת להנחה המקורית שלנו.

ראינו כי למאפיינים שונים יש השפעה על הפופולריות של מכשיר ורצינו לבנות מסווג שידע לחזות האם מכשיר בעל מאפיינים מסוימים יהיה פופולארי יותר או פחות.
בחרנו לעבוד עם המאפיינים הבאים: המחיר הטוב ביותר, מספר מוכרים, גודל מסך, גודל סוללה.
בחרנו במאפיינים אלו מסיבות שונות, בהן מגבלות שנבעו מחוסרים במסד הנתונים וממשתנים שונים בעלי קורלציות זהות.
נציין כי המאפיינים שבחרנו מושפעים על ידי הייצרן, בנוסף, אנו חושבים כי כמות המוכרים אכן מושפעת על ידי הייצרן (אפילו אם בעקיפין) מכיוון שבידיו ההחלטה עם כמה יבואנים לעבוד ובאיזו צורה למכור את המכשירים.
בחרנו לאמן מסווג באמצעות אלגוריתם k-NN
לאחר בדית CV בחרנו להשתמש בערך k=17
רמת הדיוק של המסווג היא 74%
והוא בעל ערך precision של 0.78 וrecall של 0.74
באופן כללי, המסווג דיי טוב ומאוזן בסיווג של מכשירים פופולאריים או לא פופולאריים, אך עם זאת הוא עדיין בעל רמת דיוק של כ69% שזה לא מושלם.
היינו ממליצים לחברות להשתמש במסווג הנ"ל אך להשתמש בו רק על מנת לקבל אינדיקציה ראשונית ולא לקבל החלטות משמעותיות באמצעותו.

ראינו כי במידה גבוהה ניתן לחזות פופולאריות של מכשיר על סמך מאפייניו, דבר שיכול לענות על צורך מרכזי של יצרן הפלאפונים.

מגבלות

- פיצ'רים מסוימים במסד הנתונים סובלים מחוסרים במידע, בחרנו שלא להשלים מידע זה באמצעות ערכי ממוצע או ערכים אקראיים מכיוון שלהחלטה זו השפעה גדולה מאוד על המחקר בכך שהיא יכולה לעוות את היחסים בין הפיצ'רים השונים.
התמודדנו עם החוסרים במספר דרכים:
- בפיצ'רים בהם היו חוסרים מועטים (עד 10) בחרנו להתעלם מכיוון שכמות החוסרים הייתה זניחה.
- בפיצ'רים בהם היו חוסרים רבים בחרנו להשתמש רק במקרים בהם ניתן היה להתעלם מהחוסרים בצורה אשר לא תשפיע דרמטית על המסקנות
בחישוב הקורלציה בין מחיר לבין גודל סוללה, על אף שהיו חוסרים רבים (כ 10%) בגודל הסוללה, בחרנו להתעלם מהחוסרים וקיבלנו קורלציה מספיק חזקה כך שהיא אמינה גם ללא הנתונים החסרים. כך גם בצבעי המכשירים בחרנו להתעלם מהחוסרים כי לא היה ניתן להשלים את החוסרים באף דרך, ומספר החוסרים היה מועט דיו שנוכל להסיק מסקנות בלי קשר.
- בפיצ'רים של המחיר הגבוה והמחיר הנמוך (חוסרים של כ 20%) בחרנו להתעלם מהפיצ'ר לחלוטין מכיוון שהיה לנו את המחיר הטוב ביותר שנמצא איתם בקורלציה גבוהה מאוד (כמעט 1) והשימוש בו היה אולטימטיבי.
- בבדיקת ההתפלגות של תאריך הוצאת הדגם שמנו לב לשתי תופעות




הראשונה, קטימה מימין, ניתן לראות שבשנת 2021 לא נמדדו הרבה דגמים, דבר זה נובע מכך שהנתונים קטומים מימין. ניתן לראות בהתבוננות במסד כי הדגם האחרון במסד יצא בחודש פברואר 2021 ולכן אין תיעוד למכשירים רבים בשנה זו.
השנייה, חוסר איזון בנתונים העלול להוביל להזנחת שיעור בסיס.
דבר זה נובע מכך שעם התקדמות הזמן קיימת עליה בכמות המכשירים שיצאו.
- כפי שתיארנו בתחילה, מצאנו במסד מספר כפילויות ובחרנו להסירן ולהזין את דירוג הפופולאריות הממוצע ביניהן.
בנוסף, קיימים מכשירים אשר להם רשומה נפרדת לכל צבע על אף שהמכשיר זהה, בחרנו להשאיר רשומות אלו מכיוון שבעינינו צבע שונה גורם ללוקחות להתנהג באופן שונה עם המוצר.

- המשתנה המרכזי בפרויקט שלנו היה משתנה הפופולריות, משתנה זה מחושב באופן יחסי למכשירים האחרים, כלומר, אם המכשיר הפופולרי ביותר הוא פופולרי כפליים מהמכשיר השני הפופולרי ביותר הבדל זה לא יבוא לידי ביטוי במדד הפופולריות וה"מרחק" ביניהם יישאר זהה. במסד אופטימלי היינו מעדיפים נתון של כמה מכשירים נקנו או נתון דומה ולא של דירוג הפופולריות שלהם.
בנוסף, לאחר מחיקת הכפילויות והשמת הערך הממוצע ברשומות שנשארו, ישנן הטיות נוספות במרחקים של הפופולריות.
- ראינו כי מכשירם בצע פחם מאוד פופולריים ביחד למכשירים אחרים, אך חשוב לזכור כי ישנו מספר מועט מאוד שלהם ולכן יש לקחת זאת בחשבון שאנחנו רוצים להסיק מסקנות הקשורות לצבע הזה.
- כל המסקנות שקיבלנו רלוונטיות אך ורק לאוקרינה ולפרק הזמן שבו נאספו הנתונים.
ייתכן והמסקנות תקפות גם לאזורים שונים בעולם אך על מנת לוודא זאת יש צורך בנתונים מכל העולם.

## כיוונים לעתיד

בעקבות המחקר שביצענו וכך שהנתונים מגיעים רק עד לתחילת שנת 2021, שאלת מחקר מעניינת שהיינו רוצים לברר בהמשך היא "האם הטלפונים הפופולאריים ביותר שיצאו בהמשך 2021 ועד היום הם בעלי מאפיינים דומים למאפיינים שתרמו לפופולאריות אותם מצאנו במהלך המחקר הנוכחי?"

בנוסף, מ domain knowledge שיש לנו בתחום אנחנו יודעים שהפלאפון הסלולרי החליף כמעט לחלוטין את המצלמה הביתית.
היינו שמחים לבדוק איך מערך הצילום של הפלאפון משפיע על הפופולאריות שלו לאורך השנים.
על מנת לבדוק זאת עלינו היה לאסוף מידע על המצלמות בכל דגם, מהדגמים שיש לנו במסד (גודל חיישן, סוג עדשה, מפתח צמצם רזולוציה וכו').
על סמך נתונים אלו היינו משקללים ציון למערך הצילום ובודקים את הקשר בינו לבין הפופולאריות לאורך השנים.