Final Dataset Report

June 4, 2021

1 Dataset report

by Robin Fettelaar and Vinzenz Richard Ulrich

1.1 Abstract

Computers play an important role in our daily lives, and a new computer is an investment that you want to optimally utilize for years to come. Since the computer industry is volatile it is useful for potential buyers and sellers to know what the standard price for a given computer would be, to see if they are getting a good deal. If a buyer finds a computer that costs less than the normal price with the specifications of the computer, the buyer is saving money; if a seller finds out the price that is asked for a product is actually lower than the normal price, the seller can increase the price and make more money. A way to find out what the standard/normal price for a computer with given specifications is is thus helpful for both parties. For that, we have designed two linear models by means of multiple regression which are able to predict the price of a computer given the harddrive size, the RAM size, the screen size, whether or not the computer has a CD, multiple ports and whether or not the computer is premium. Whether the computer has multiple ports appeared to be insignificant in predicting the price of a computer.

1.2 Introduction

In today's world, having a computer is indespensable. You might use it for your work, for school, or just to play video games. However, there are a lot of overpriced computers being sold today, especially with the ongoing chip shortage, which has lead to a general increase in prices. Overpaying for a computer is annoying and entirely preventable. Conversely, computer manufacturers want to avoid asking for too little money and in the end making a lower profit than was possible. There are many things that influence the price of a computer, and we want to look at how some of those interact with the price in this report. This may give consumers insights into whether or not the computer they are considering buying is too expensive and gives manufacturers insights into whether or not they are charging less than they could.

Our research question is thus: "Do speed, the size of the harddrive, RAM and screen, whether or not the computer has a CD player, whether or not the computer has multiple ports, and whether it is premium influence the price of a computer?"

In order to answer this question, we have constructed two multiple regression models which explain which of the factors have an association with the price of a computer. The models also show whether the association is positive or negative. In the first model, the speed, RAM size and screen size are positively associated with the price, while the harddrive space, the computer being premium and the computer having a CD player are negatively associated with the price. In the second model, where we only took the 359 most recent data points, however, all explanatory variables are positively associated with price, except the computer being premium, which is negatively associated with price. There is no special reason why we chose the value 359 besides that it seemed a good amount to represent the more recent computers. The second model is purely used to gain a better insight of the data and will not be used to draw conclusions from.

1.2.1 Hypotheses

H_0: The speed of the computer, the size of the harddrive, the size of the RAM installed, the size of the screen, whether or not a CD player is installed, whether or not there are multiple ports on the computer and whether or not the computer is premium do not influence the price of a computer.

H_A: The speed of the computer, the size of the harddrive, the size of the RAM installed, the size of the screen, whether or not a CD player is installed, whether or not there are multiple ports on the computer and whether or not the computer is premium influence the price of a computer.

1.3 Data

The dataset we are using is the Basic Computer Data Dataset from Kaggle. This dataset consists of 6259 observations from computers. The variables in this dataset are: price, speed, hd, ram, screen, cd, multi, premium.

1.3.1 Population

Observational units are: dollars for the price, hard drive in gigabyte for hd, ram size in gigabyte for ram, screen size in inches for the screen size. The other variables don't have a measurement unit. The data can be used to draw conclusions about future computer prices.

1.3.2 Response Variable

We want to be able to predict the price of computers. The price is measured in dollars. In order to predict the price, we will use the continuous variable price, which consists of approximately normally distributed values which range from 949 up to 5399, as the response variable.

1.3.3 Explanatory Variables

The speed can have an influence, because people generally want faster computers which could be associated with the price. The speed is a discrete categorical variable that can consist of the values 25, 33, 50, 66, 75 or 100.

The size of the hard drive (hd) can influence the response variable because when computers have a larger hard drive, more files can be stored on the computer. hd is a cardinal variable which can take values from 80 up to 2100.

The size of the RAM that is installed, which is number of gigabytes, could also be influential. ram is a discrete categorical value which can consist of the values 2, 4, 8, 16, 24 or 32.

The size of the screen is another explanatory variable because some screen sizes can be more popular among potential buyers, which would lead to an increase in price. The screen size is a discrete categorical value which can consist of the values 14, 15 or 17.

Whether the computer has a CD player is another potentially important explanatory variable, because some people might really need one, and are willing to pay significantly more for a computer that has one, or not having a CD player might be associated with the computer being 'sleek' and 'modern', which could drive the price up. cd is a categorical binary variable which can take the values 0 (the computer does not have a CD player) or 1 (the computer does have a CD player).

Whether the computer has multiple ports could also be an important explanatory variable due to the fact that computer users might want to use multiple ports and thus are willing to pay a higher price for this. Whether the computer is premium or not can be influential because if the computer seems to be premium, potential buyers could be willing to pay more for the computer. Premium is a categorical binary variable whch can take the values 0 (The computer is premimum) or 1 (the computer is not premium)

1.4 Analysis of the variables

[1]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import statsmodels.api as sm
from scipy.stats import probplot
import warnings
warnings.filterwarnings('ignore')
```

[2]:
```
computer_prices = pd.read_csv('Computers.csv')
```

[3]:
```
dummy_coding = {'2': 0, '4': 1, '8': 2, '16': 3, '24': 4, '32': 5}
ram_dummy = computer_prices['ram'].copy()
ram_dummy = ram_dummy.replace(dummy_coding)
computer_prices['ram_dummy']=ram_dummy

dummy_coding2 = {'14': 0, '15': 1, '17': 2}
screen_dummy = computer_prices['screen'].copy()
screen_dummy = screen_dummy.replace(dummy_coding2)
computer_prices['screen_dummy']=screen_dummy

dummy_coding3 = {'no': 0, 'yes': 1}
cd_dummy = computer_prices['cd'].copy()
cd_dummy = cd_dummy.replace(dummy_coding3)
computer_prices['cd_dummy']=cd_dummy

dummy_coding4 = {'25': 0, '33': 1, '50': 2, '66': 3, '75': 4, '100': 5}
speed_dummy = computer_prices['speed'].copy()
speed_dummy = speed_dummy.replace(dummy_coding4)
computer_prices['speed_dummy']=speed_dummy
```

```
dummy_coding5 = {'no': 0, 'yes': 1}
premium_dummy = computer_prices['premium'].copy()
premium_dummy = premium_dummy.replace(dummy_coding5)
computer_prices['premium_dummy']=premium_dummy

dummy_coding6 = {'no': 0, 'yes': 1}
multi_dummy = computer_prices['multi'].copy()
multi_dummy = multi_dummy.replace(dummy_coding6)
computer_prices['multi_dummy']=multi_dummy
```

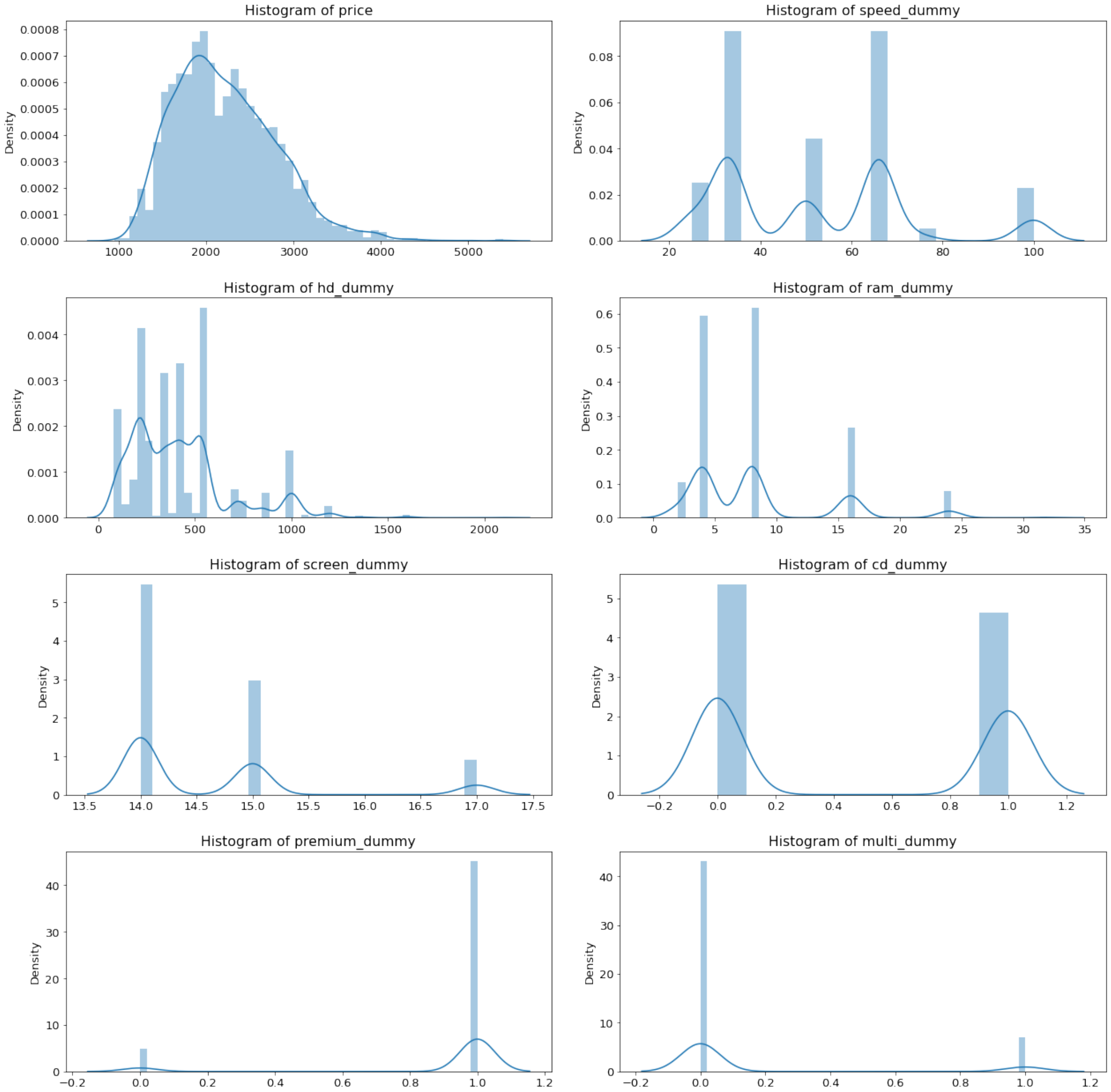
Before fitting the model, it is important to have a feeling of how the variables in our dataset are distributed. Below, the distribution of each of the variables is shown.

[20]:

```
fig, ax = plt.subplots(4, 2, figsize=(18, 18))
fig.tight_layout(pad = 3)
plt.rcParams.update({'font.size': 13})

sns.distplot(ax = ax[0][0], x = computer_prices['price'])
ax[0][0].set_title("Histogram of price")

sns.distplot(ax = ax[0][1], x = computer_prices['speed_dummy'])
ax[0][1].set_title("Histogram of speed_dummy")

sns.distplot(ax = ax[1][0], x = computer_prices['hd'])
ax[1][0].set_title("Histogram of hd_dummy")

sns.distplot(ax = ax[1][1], x = computer_prices['ram_dummy'])
ax[1][1].set_title("Histogram of ram_dummy")

sns.distplot(ax = ax[2][0], x = computer_prices['screen_dummy'])
ax[2][0].set_title("Histogram of screen_dummy")

sns.distplot(ax = ax[2][1], x = computer_prices['cd_dummy'])
ax[2][1].set_title("Histogram of cd_dummy")

sns.distplot(ax = ax[3][0], x = computer_prices['premium_dummy'])
ax[3][0].set_title("Histogram of premium_dummy")

sns.distplot(ax = ax[3][1], x = computer_prices['multi_dummy'])
ax[3][1].set_title("Histogram of multi_dummy")
```

[20]:

```
Text(0.5, 1.0, 'Histogram of multi_dummy')
```



In order to predict the price, we can make use of the seven potentially influential variables. To find out which variables will best predict the price, we used backward-selection based on the p-values. When we fitted a linear model including all explanatory variables, there appeared to be only one variable which was insignificant, namely the variable “multi”. Refitting on the remaining explanatory variables gave us a model where all the values had a significalt p-value. Remarkable was that none of the p-values changed when removing the variable multi from the model.

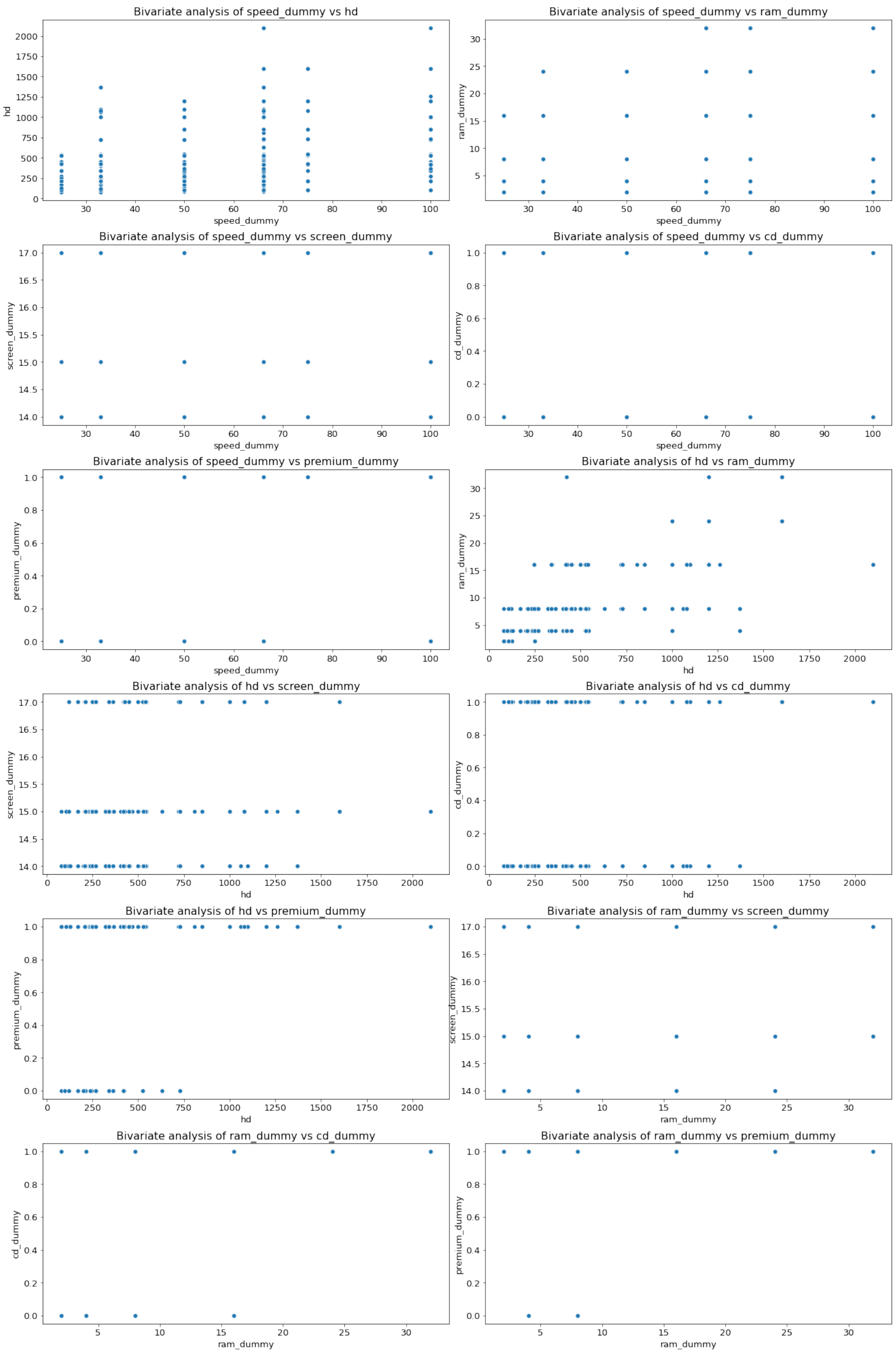
```
[5]: m_full = sm.formula.ols(formula = 'price ~ speed_dummy + hd + ram_dummy +␣
      ↪screen_dummy + cd_dummy + premium_dummy', data = computer_prices)
     multi_reg = m_full.fit()
     print(multi_reg.summary())
```

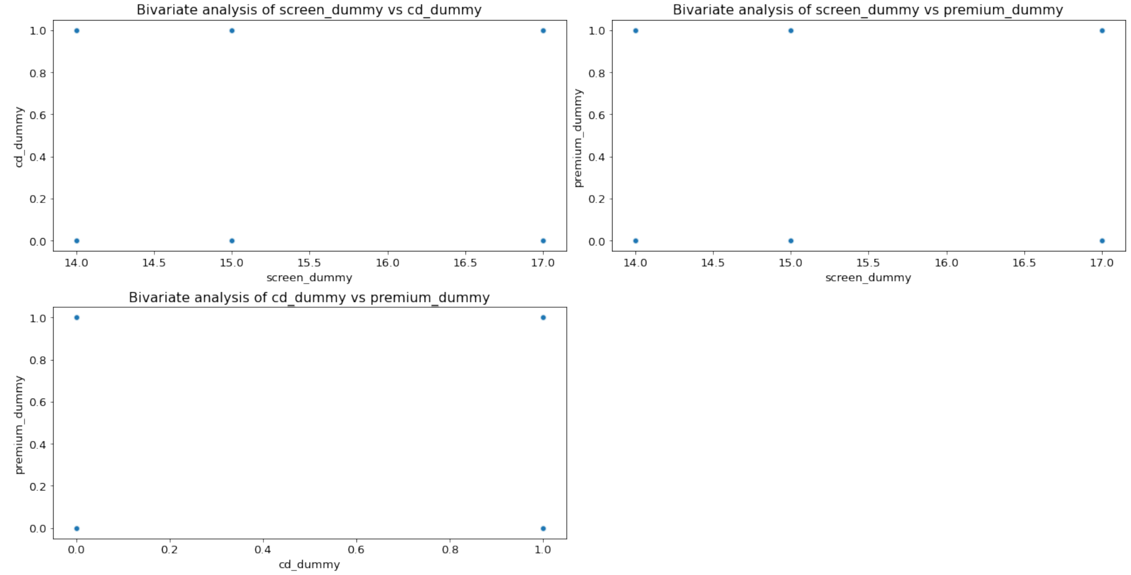
```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.504
Model:                            OLS   Adj. R-squared:                  0.504
Method:                 Least Squares   F-statistic:                     1060.
Date:                Fri, 04 Jun 2021   Prob (F-statistic):               0.00
Time:                        23:08:15   Log-Likelihood:                -46519.
No. Observations:                6259   AIC:                         9.305e+04
Df Residuals:                    6252   BIC:                         9.310e+04
Df Model:                           6
Covariance Type:            nonrobust
================================================================================
=
                    coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
-
Intercept       386.6357     86.147      4.488      0.000     217.758
555.514
speed_dummy       5.7470      0.268     21.451      0.000       5.222
6.272
hd               -0.4849      0.035    -13.994      0.000      -0.553
-0.417
ram_dummy        80.2050      1.480     54.200      0.000      77.304
83.106
screen_dummy    100.4178      5.927     16.942      0.000      88.799
112.037
cd_dummy        -75.8481     12.204     -6.215      0.000     -99.772
-51.924
premium_dummy  -399.3892     17.989    -22.202      0.000    -434.654
-364.124
==============================================================================
Omnibus:                     1355.726   Durbin-Watson:                   1.443
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             4158.703
Skew:                           1.109   Prob(JB):                         0.00
Kurtosis:                       6.320   Cond. No.                     8.24e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 8.24e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

It is important that there is no colinearity among the each of the variables, therefore a bivariate analysis is made to visualize the correlation between each of the two explanatory variables. Strongly correlated variables can influence the result of the model strongly depending on which variables are included first, hence it is important to take the colinearity into account for the interpretation of the model.

[28]:

```
fig2, ax2 = plt.subplots(6, 2, figsize =(18, 27))
fig2.tight_layout(h_pad = 3)
ax2 = ax2.flatten()
plt.rcParams.update({'font.size': 13})

first_variable = ['speed_dummy', 'speed_dummy', 'speed_dummy', 'speed_dummy',␣
 ↪'speed_dummy', 'hd', 'hd', 'hd', 'hd', 'ram_dummy', 'ram_dummy', 'ram_dummy']
second_variable = ['hd', 'ram_dummy', 'screen_dummy', 'cd_dummy',␣
 ↪'premium_dummy', 'ram_dummy', 'screen_dummy', 'cd_dummy', 'premium_dummy',␣
 ↪'screen_dummy', 'cd_dummy', 'premium_dummy']
#We compare each of the variables against each other in a scatterplot
for index in range(len(first_variable)):
    sns.scatterplot(computer_prices[first_variable[index]],␣
 ↪computer_prices[second_variable[index]], ax = ax2[index])
    ax2[index].set_title('Bivariate analysis of {} vs {}'.
 ↪format(first_variable[index], second_variable[index]))


fig3, ax3 = plt.subplots(2, 2, figsize =(18, 9))
fig3.tight_layout(h_pad = 3)
ax3 = ax3.flatten()
plt.rcParams.update({'font.size': 13})

first_variable2 = ['screen_dummy', 'screen_dummy', 'cd_dummy']
second_variable2 = ['cd_dummy', 'premium_dummy', 'premium_dummy']
#We make two loops to improve the way that the plots are rendered in the PDF file
for index in range(len(first_variable2)):
    sns.scatterplot(computer_prices[first_variable2[index]],␣
 ↪computer_prices[second_variable2[index]], ax = ax3[index])
    ax3[index].set_title('Bivariate analysis of {} vs {}'.
 ↪format(first_variable2[index], second_variable2[index]))

ax3[3].set_visible(False)
```


Bivariate analysis of speed_dummy vs hd
hd
speed_dummy
0
250
500
750
1000
1250
1500
1750
2000
30
40
50
60
70
80
90
100
Bivariate analysis of speed_dummy vs ram_dummy
ram_dummy
speed_dummy
5
10
15
20
25
30
Bivariate analysis of speed_dummy vs screen_dummy
screen_dummy
speed_dummy
14.0
14.5
15.0
15.5
16.0
16.5
17.0
Bivariate analysis of speed_dummy vs cd_dummy
cd_dummy
speed_dummy
0.0
0.2
0.4
0.6
0.8
1.0
Bivariate analysis of speed_dummy vs premium_dummy
premium_dummy
speed_dummy
Bivariate analysis of hd vs ram_dummy
ram_dummy
hd
Bivariate analysis of hd vs screen_dummy
screen_dummy
hd
Bivariate analysis of hd vs cd_dummy
cd_dummy
hd
Bivariate analysis of hd vs premium_dummy
premium_dummy
hd
Bivariate analysis of ram_dummy vs screen_dummy
screen_dummy
ram_dummy
Bivariate analysis of ram_dummy vs cd_dummy
cd_dummy
ram_dummy
Bivariate analysis of ram_dummy vs premium_dummy
premium_dummy
ram_dummy



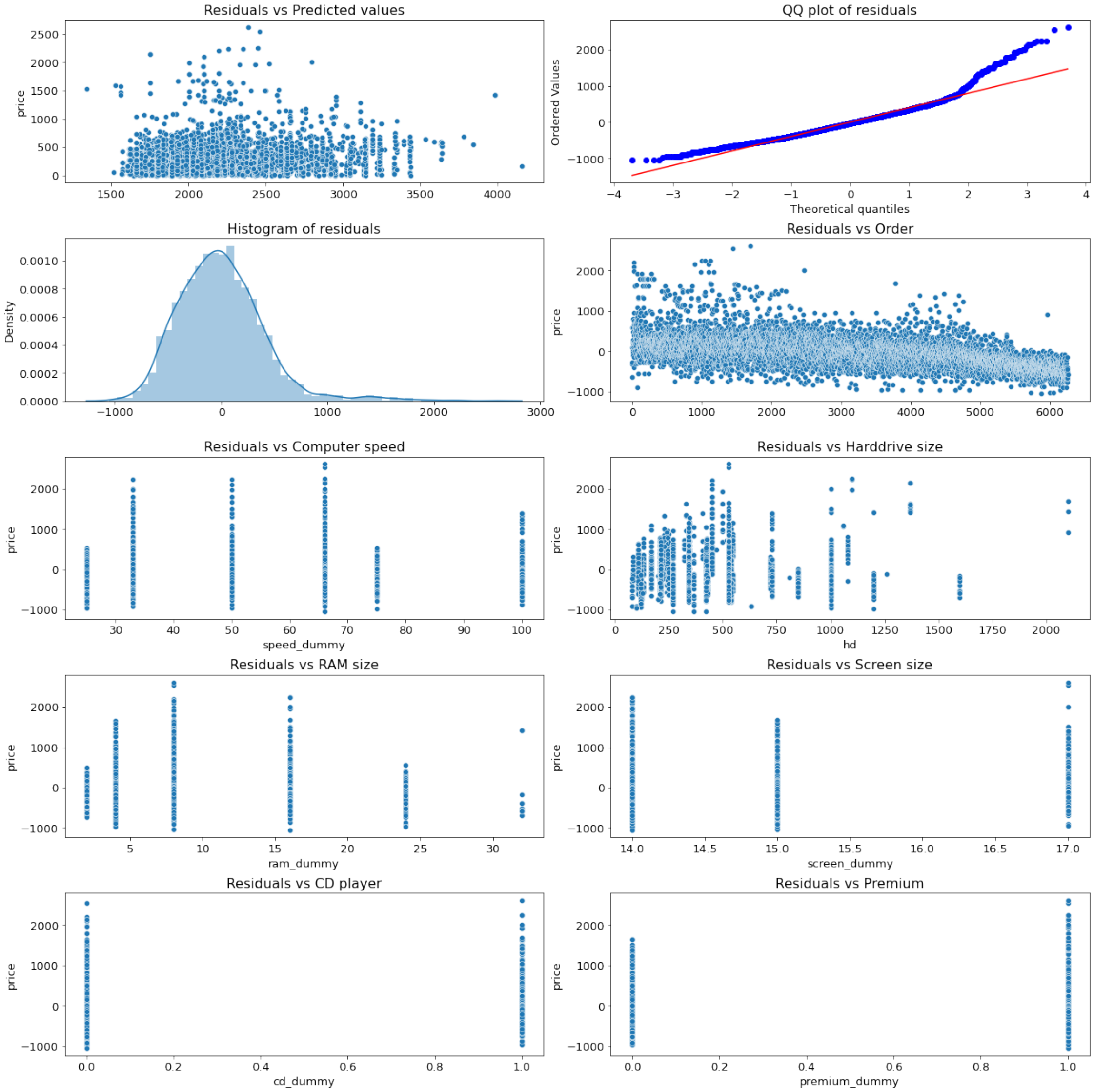
There appears to be no collinearity between most of the variables. However, there appears to be a very weak correlation between speed_dummy and hd, aswell as hd vs ram_dummy and possibily speed_dummy vs ram_dummy. However, this correlations seem to be very weak if even apparent, and thus, we expect this correlation to have very little to no impact on our model.

[7]:

```
predicted_value = multi_reg.predict()

residuals = computer_prices['price'] - predicted_value

fig4, ax4 = plt.subplots(5, 2, figsize=(18, 18))
fig4.tight_layout(pad = 3)
plt.rcParams.update({'font.size': 13})

sns.scatterplot(ax = ax4[0][0], x = predicted_value, y = abs(residuals))
ax4[0][0].set_title("Residuals vs Predicted values")

probplot(x = residuals, plot = ax4[0][1])
ax4[0][1].set_title("QQ plot of residuals")

sns.distplot(ax = ax3[1][0], x = residuals)
ax3[1][0].set_title("Histogram of residuals")

sns.scatterplot(ax = ax3[1][1], x = range(len(computer_prices)), y = residuals)
ax3[1][1].set_title("Residuals vs Order")
```

```
sns.scatterplot(ax = ax3[2][0], x = computer_prices['speed_dummy'], y =␣
 ↪residuals)
ax3[2][0].set_title("Residuals vs Computer speed")

sns.scatterplot(ax = ax3[2][1], x = computer_prices['hd'], y = residuals)
ax3[2][1].set_title("Residuals vs Harddrive size")

sns.scatterplot(ax = ax3[3][0], x = computer_prices['ram_dummy'], y = residuals)
ax3[3][0].set_title("Residuals vs RAM size")

sns.scatterplot(ax = ax3[3][1], x = computer_prices['screen_dummy'], y =␣
 ↪residuals)
ax3[3][1].set_title("Residuals vs Screen size")

sns.scatterplot(ax = ax3[4][0], x = computer_prices['cd_dummy'], y = residuals)
ax3[4][0].set_title("Residuals vs CD player")

sns.scatterplot(ax = ax3[4][1], x = computer_prices['premium_dummy'], y =␣
 ↪residuals)
ax3[4][1].set_title("Residuals vs Premium")
```

```
[7]: Text(0.5, 1.0, 'Residuals vs Premium')
```



The variability of the residuals seems to be constant, although there are some outliers on the left-hand side.

The residuals do seem to be approximately normally distributed, but there is a clear right-skew in the data. On the QQ-plot you can see that the values deviate from the line at the right-tale quite a lot and the right-skew is also clearly apparent in the histogram. Since we have a large sample size of 6259 observations the skew will not cause any issues because the skew will not have a large impact on the results.

The residuals seem to be mostly independent, but there is a slight downward trend over time. This could potentially be explained by the fact that computers may have gotten cheaper over time.

The residuals vs the computer speed seems to be linear, because the points do not show any clear association.

The residuals vs the hard drive size seems to be mostly linear, but there is a lot of variability.

The residuals vs the RAM size seems to be linear.

The residuals vs the screen size seems to be linear, because there is no apparent association, but it is hard to verify this, due to the fact that the screen size only takes on 3 different values.

The residuals vs CD player seems to be linear, as the ranges in which the values lie are about the same

The residuals vs premium seems to be mostly linear, although the values for the computer not being premium fall within a smaller range.

Since we suspected the old computers in the dataset may have a very large impact on the result of the multiple regression model, leading to a smaller harddrive space being associated with a higher price, we fitted a second multiple regression only considering the last 359 observations to confirm this suspicion. We think this model might be more precise for predicting the price of modern computers, and thus have moved it from the Results section, where we had it before and used it to confirm our suspicion, to the Analysis of the variables section in order to further explore it.

1.4.1 Second model

[8]:
```
computer_prices2 = pd.read_csv('Computers.csv')
computer_prices2 = computer_prices2[computer_prices2['row'] > 5900]
```

[9]:
```
dummy_coding7 = {'2': 0, '4': 1, '8': 2, '16': 3, '24': 4, '32': 5}
ram_dummy2 = computer_prices2['ram'].copy()
ram_dummy2 = ram_dummy2.replace(dummy_coding7)
computer_prices2['ram_dummy2']=ram_dummy2

dummy_coding8 = {'14': 0, '15': 1, '17': 2}
screen_dummy2 = computer_prices2['screen'].copy()
screen_dummy2 = screen_dummy2.replace(dummy_coding8)
computer_prices2['screen_dummy2']=screen_dummy2

dummy_coding9 = {'no': 0, 'yes': 1}
cd_dummy2 = computer_prices2['cd'].copy()
cd_dummy2 = cd_dummy2.replace(dummy_coding9)
computer_prices2['cd_dummy2']=cd_dummy2

dummy_coding10 = {'25': 0, '33': 1, '50': 2, '66': 3, '75': 4, '100': 5}
speed_dummy2 = computer_prices2['speed'].copy()
speed_dummy2 = speed_dummy2.replace(dummy_coding10)
computer_prices2['speed_dummy2']=speed_dummy2

dummy_coding11 = {'no': 0, 'yes': 1}
premium_dummy2 = computer_prices2['premium'].copy()
premium_dummy2 = premium_dummy2.replace(dummy_coding11)
computer_prices2['premium_dummy2']=premium_dummy2
```

```
dummy_coding12 = {'no': 0, 'yes': 1}
multi_dummy2 = computer_prices2['multi'].copy()
multi_dummy2 = multi_dummy2.replace(dummy_coding12)
computer_prices2['multi_dummy2']=multi_dummy2
```

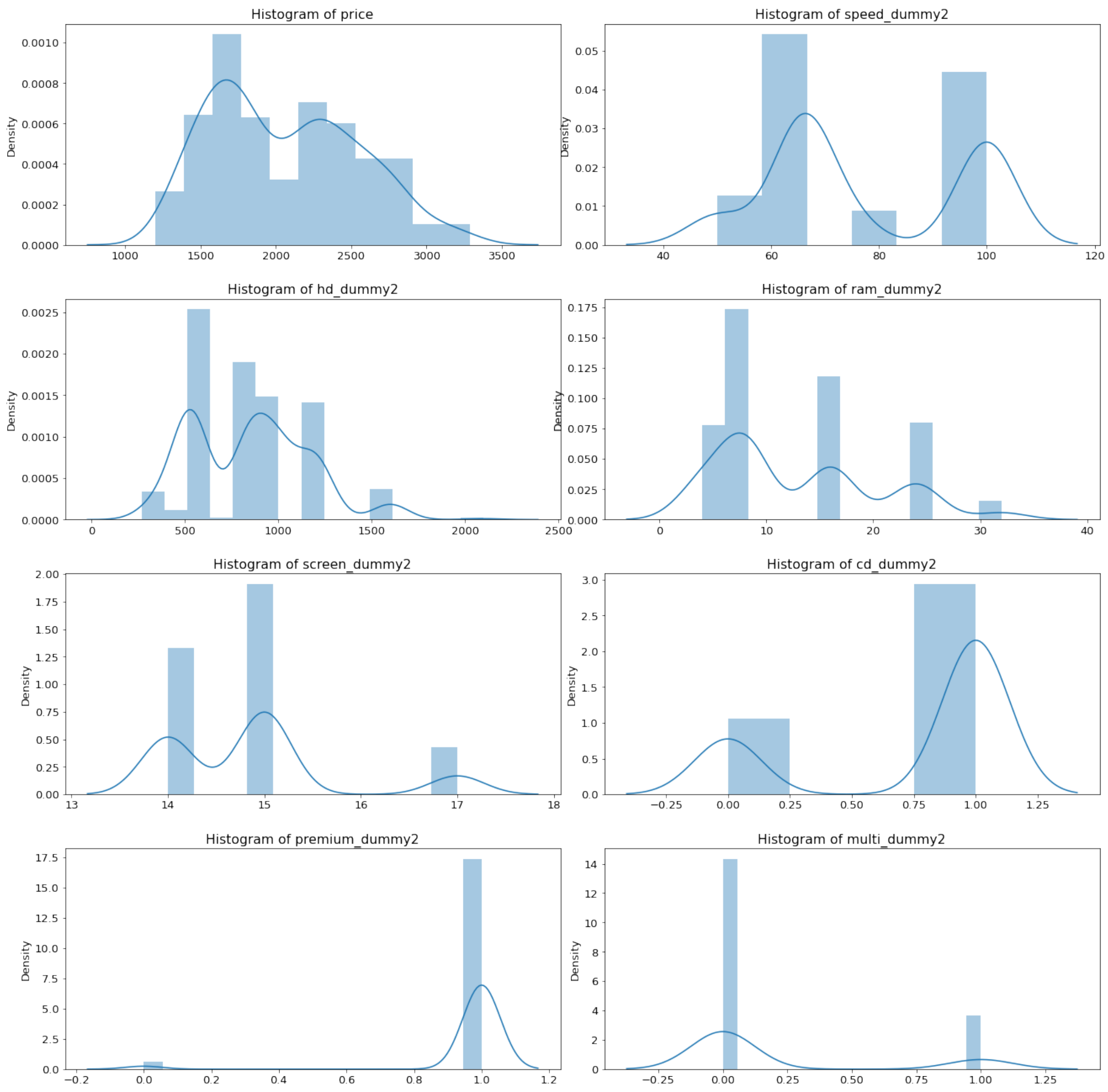
Again, it is important to have a feeling of how the variables in our dataset are distributed. Below, the distribution of each of the variables is shown.

[10]:

```
fig4, ax4 = plt.subplots(4, 2, figsize=(18, 18))
fig4.tight_layout(h_pad = 3)
plt.rcParams.update({'font.size': 13})

sns.distplot(ax = ax4[0][0], x = computer_prices2['price'])
ax4[0][0].set_title("Histogram of price")

sns.distplot(ax = ax4[0][1], x = computer_prices2['speed_dummy2'])
ax4[0][1].set_title("Histogram of speed_dummy2")

sns.distplot(ax = ax4[1][0], x = computer_prices2['hd'])
ax4[1][0].set_title("Histogram of hd_dummy2")

sns.distplot(ax = ax4[1][1], x = computer_prices2['ram_dummy2'])
ax4[1][1].set_title("Histogram of ram_dummy2")

sns.distplot(ax = ax4[2][0], x = computer_prices2['screen_dummy2'])
ax4[2][0].set_title("Histogram of screen_dummy2")

sns.distplot(ax = ax4[2][1], x = computer_prices2['cd_dummy2'])
ax4[2][1].set_title("Histogram of cd_dummy2")

sns.distplot(ax = ax4[3][0], x = computer_prices2['premium_dummy2'])
ax4[3][0].set_title("Histogram of premium_dummy2")

sns.distplot(ax = ax4[3][1], x = computer_prices2['multi_dummy2'])
ax4[3][1].set_title("Histogram of multi_dummy2")

#axs5[3][1].set_visible(False)
```

[10]:

```
Text(0.5, 1.0, 'Histogram of multi_dummy2')
```



Again, we will use backward-selection based on the p-values. When we fit a linear model including all explanatory variables, there appeared to be only one variable which was insignificant, namely the variable "multi". Refitting on the remaining explanatory variables gave us a model where all the values had a significalt p-value. Remarkable was that none of the p-values changed when removing the variable multi from the model.

[11]:
```
print("The model of just the last 359 data points:")
m_full2 = sm.formula.ols(formula = 'price ~ speed_dummy2 + hd + ram_dummy2 +␣
 ↪screen_dummy2 + cd_dummy2 + premium_dummy2', data = computer_prices2)
multi_reg2 = m_full2.fit()
print(multi_reg2.summary())
```

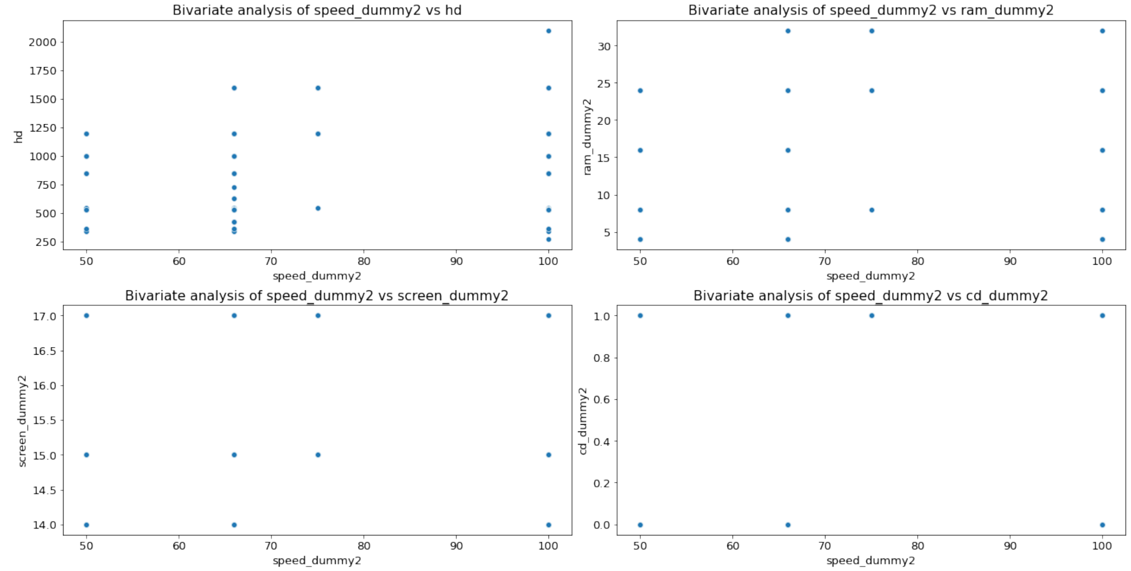
```
The model of just the last 359 data points:
                            OLS Regression Results
```

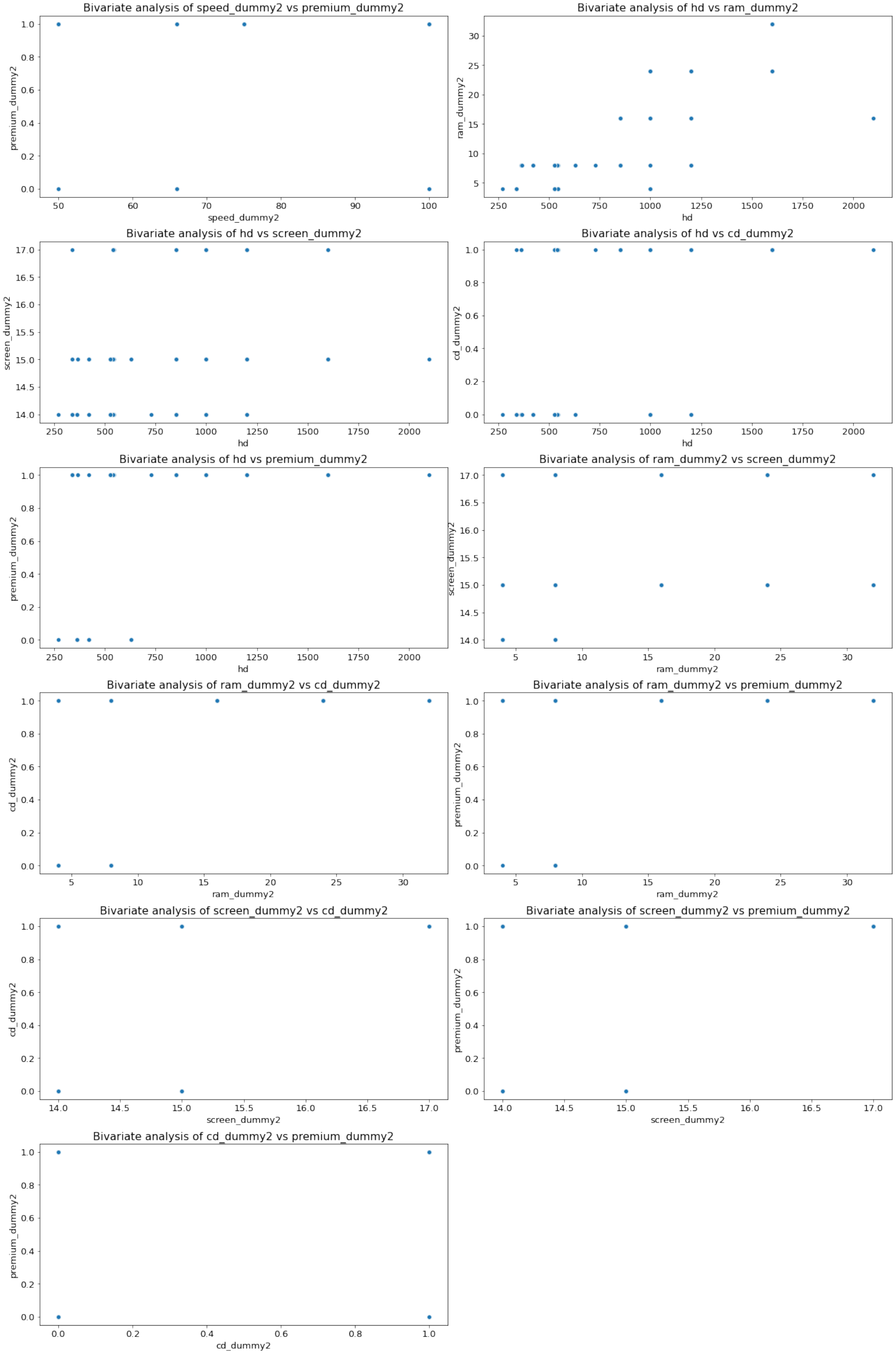
```
==============================================================================
Dep. Variable:                  price   R-squared:                       0.942
Model:                            OLS   Adj. R-squared:                  0.941
Method:                 Least Squares   F-statistic:                     950.6
Date:                Fri, 04 Jun 2021   Prob (F-statistic):          4.75e-214
Time:                        23:08:23   Log-Likelihood:                -2218.9
No. Observations:                 359   AIC:                             4452.
Df Residuals:                     352   BIC:                             4479.
Df Model:                           6
Covariance Type:            nonrobust
================================================================================
==
                     coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
--
Intercept       -134.9973    118.733     -1.137      0.256    -368.512
98.517
speed_dummy2       3.7212      0.347     10.714      0.000       3.038
4.404
hd                 0.2200      0.038      5.825      0.000       0.146
0.294
ram_dummy2        43.9818      1.620     27.155      0.000      40.796
47.167
screen_dummy2     94.1987      7.827     12.036      0.000      78.806
109.592
cd_dummy2        120.0448     18.040      6.654      0.000      84.566
155.524
premium_dummy2  -350.4067     36.974     -9.477      0.000    -423.124
-277.689
==============================================================================
Omnibus:                       27.709   Durbin-Watson:                   1.720
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               87.947
Skew:                           0.242   Prob(JB):                     7.99e-20
Kurtosis:                       5.376   Cond. No.                     1.74e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 1.74e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

[38]:

```python
fig5, ax5 = plt.subplots(2, 2, figsize =(18, 9))
fig5.tight_layout(h_pad = 3)
ax5 = ax5.flatten()
plt.rcParams.update({'font.size': 13})
```

```
first_variable3 = ['speed_dummy2', 'speed_dummy2', 'speed_dummy2',␣
 ↪'speed_dummy2']
second_variable3 = ['hd', 'ram_dummy2', 'screen_dummy2', 'cd_dummy2']
#We compare each of the variables against each other in a scatterplot
for index in range(len(first_variable3)):
    sns.scatterplot(computer_prices2[first_variable3[index]],␣
 ↪computer_prices2[second_variable3[index]], ax = ax5[index])
    ax5[index].set_title('Bivariate analysis of {} vs {}'.
 ↪format(first_variable3[index], second_variable3[index]))


fig6, ax6 = plt.subplots(6, 2, figsize =(18, 27))
fig6.tight_layout(h_pad = 3)
ax6 = ax6.flatten()
plt.rcParams.update({'font.size': 13})

first_variable4 = ['speed_dummy2', 'hd', 'hd', 'hd', 'hd', 'ram_dummy2',␣
 ↪'ram_dummy2', 'ram_dummy2', 'screen_dummy2', 'screen_dummy2', 'cd_dummy2']
second_variable4 = ['premium_dummy2', 'ram_dummy2', 'screen_dummy2',␣
 ↪'cd_dummy2', 'premium_dummy2', 'screen_dummy2', 'cd_dummy2', 'premium_dummy2',␣
 ↪'cd_dummy2', 'premium_dummy2', 'premium_dummy2']
#We make two loops to improve the way that the plots are rendered in the PDF file
for index in range(len(first_variable4)):
    sns.scatterplot(computer_prices2[first_variable4[index]],␣
 ↪computer_prices2[second_variable4[index]], ax = ax6[index])
    ax6[index].set_title('Bivariate analysis of {} vs {}'.
 ↪format(first_variable4[index], second_variable4[index]))

ax6[11].set_visible(False)
```




Bivariate analysis of speed_dummy2 vs premium_dummy2
premium_dummy2
0.0
0.2
0.4
0.6
0.8
1.0
speed_dummy2
50
60
70
80
90
100
Bivariate analysis of hd vs ram_dummy2
ram_dummy2
5
10
15
20
25
30
hd
250
500
750
1000
1250
1500
1750
2000
Bivariate analysis of hd vs screen_dummy2
screen_dummy2
14.0
14.5
15.0
15.5
16.0
16.5
17.0
hd
Bivariate analysis of hd vs cd_dummy2
cd_dummy2
hd
Bivariate analysis of hd vs premium_dummy2
premium_dummy2
hd
Bivariate analysis of ram_dummy2 vs screen_dummy2
screen_dummy2
ram_dummy2
Bivariate analysis of ram_dummy2 vs cd_dummy2
cd_dummy2
ram_dummy2
Bivariate analysis of ram_dummy2 vs premium_dummy2
premium_dummy2
ram_dummy2
Bivariate analysis of screen_dummy2 vs cd_dummy2
cd_dummy2
screen_dummy2
Bivariate analysis of screen_dummy2 vs premium_dummy2
premium_dummy2
screen_dummy2
Bivariate analysis of cd_dummy2 vs premium_dummy2
premium_dummy2
cd_dummy2
0.0
0.2
0.4
0.6
0.8
1.0

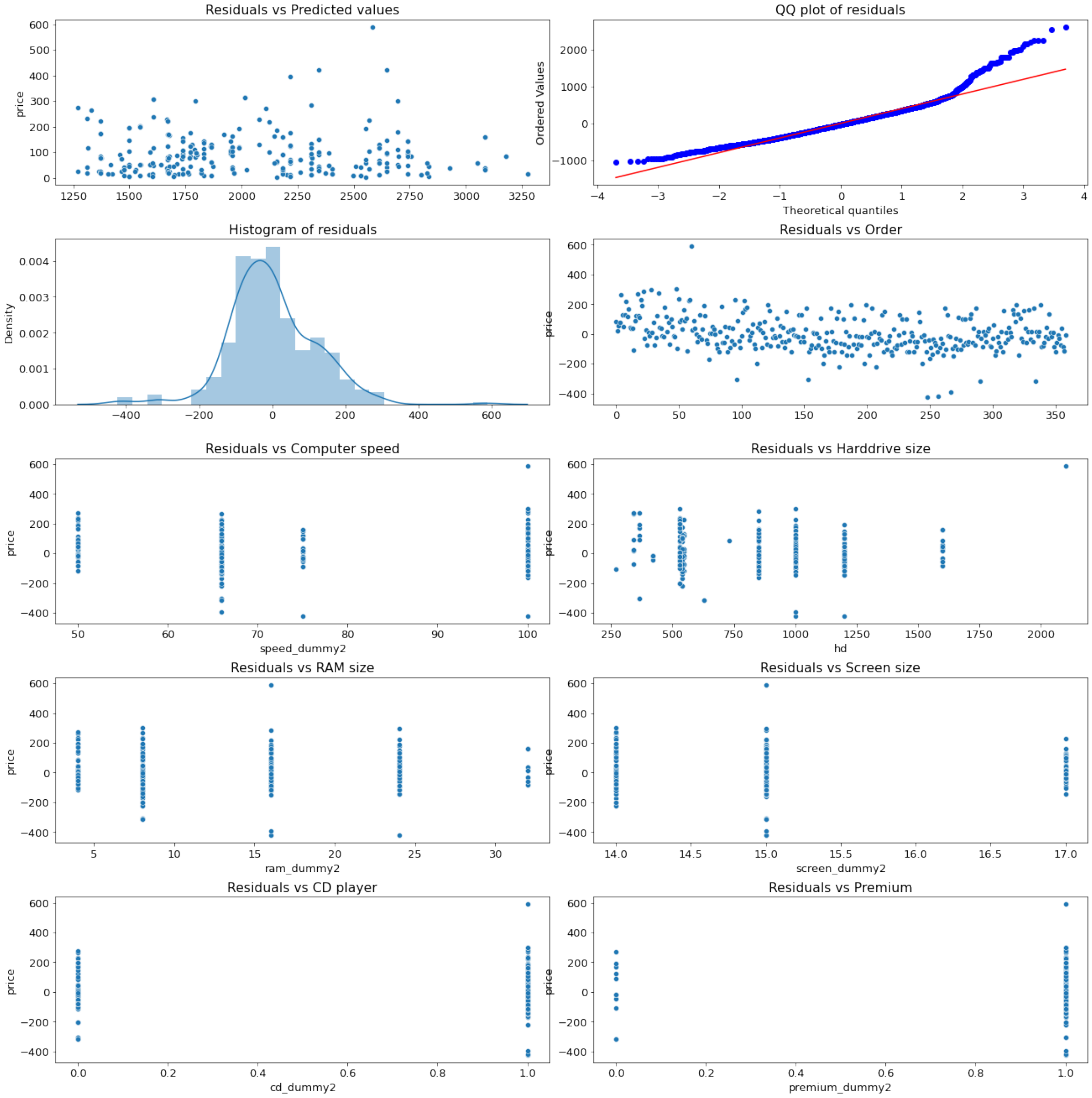
There does not appear to be collinearity between any variables. However, there does seem to be some correlation between hd and ram_dummy2, but since the correlation is pretty low, we don't think this has a significant effect on the model. Very small correlations also seem to be visible for hd vs speed_dummy2 and speed_dummy2 vs ram_dummy2.

[13]:

```python
predicted_value2 = multi_reg2.predict()

residuals2 = computer_prices2['price'] - predicted_value2

fig6, ax6 = plt.subplots(5, 2, figsize=(18, 18))
fig6.tight_layout(h_pad = 3)
plt.rcParams.update({'font.size': 13})

sns.scatterplot(ax = ax6[0][0], x = predicted_value2, y = abs(residuals2))
ax6[0][0].set_title("Residuals vs Predicted values")

probplot(x = residuals, plot = ax6[0][1])
ax6[0][1].set_title("QQ plot of residuals")

sns.distplot(ax = ax6[1][0], x = residuals2)
ax6[1][0].set_title("Histogram of residuals")

sns.scatterplot(ax = ax6[1][1], x = range(len(computer_prices2)), y = residuals2)
ax6[1][1].set_title("Residuals vs Order")

sns.scatterplot(ax = ax6[2][0], x = computer_prices2['speed_dummy2'], y =␣
 ↪residuals2)
ax6[2][0].set_title("Residuals vs Computer speed")

sns.scatterplot(ax = ax6[2][1], x = computer_prices2['hd'], y = residuals2)
ax6[2][1].set_title("Residuals vs Harddrive size")

sns.scatterplot(ax = ax6[3][0], x = computer_prices2['ram_dummy2'], y =␣
 ↪residuals2)
ax6[3][0].set_title("Residuals vs RAM size")

sns.scatterplot(ax = ax6[3][1], x = computer_prices2['screen_dummy2'], y =␣
 ↪residuals2)
ax6[3][1].set_title("Residuals vs Screen size")

sns.scatterplot(ax = ax6[4][0], x = computer_prices2['cd_dummy2'], y =␣
 ↪residuals2)
ax6[4][0].set_title("Residuals vs CD player")
```

```
sns.scatterplot(ax = ax6[4][1], x = computer_prices2['premium_dummy2'], y =␣
 ↪residuals2)
ax6[4][1].set_title("Residuals vs Premium")
```

```
[13]: Text(0.5, 1.0, 'Residuals vs Premium')
```



The variability of the residuals seems to be constant, although there appears to be a slight decrease in variability on the right side.

The residuals do seem to be approximately normally distributed, but there is a clear right-skew in the data. On the QQ-plot you can see that the values deviate from the line at the right-tale quite a lot and the right-skew is also clearly apparent in the histogram. Due to the fact that our sample size is still pretty large, the skew will not have a large impact on the results.

The residuals seem to be independent, the downward trend that was apparent in the model of the full dataset is not apparent anymore. This could possibly be explained by the fact that changes in the computer industry play less of a role because the observations lay in a closer interval.

The residuals vs the computer speed seems to be linear, because the points do not show any clear association.

The residuals vs the hard drive size seems to be mostly linear, but it is important to note that there is a large outlier at the top right.

The residuals vs the RAM size seems to be linear. There does not appear to be a clear association and the variability seems reasonably constant.

The residuals vs the screen size seems to be linear, and there is no association apparent, however, it is hard to verify this, because the screen size can only take on 3 different values.

The residuals vs CD player seems to be linear, as the ranges in which the values lay are about the same

The residuals vs premium seems to be mostly linear, although the values for the computer not being premium fall within a smaller range.

1.5 Results

Our *first model* led us to the following linear equation:

$price = 386.64 + 5.75 * speed\_dummy - 0.48 * hd + 80.21 * ram\_dummy + 100.42 * screen\_dummy - 75.85 * cd\_dummy - 399.39 * premium\_dummy$

For an increase of 1 in speed, the predicted price increases with 5.75 dollars.

For an increase of 1 in hd (hard drive size), the predicted price decreases with -0.48 dollars.

For an increase of 1 in ram (RAM size), the predicted price increases with 80.21 dollars.

For an increase of 1 in screen (screen size), the predicted price increases with 100.42 dollars.

When the computer has a cd player, the predicted price decreases with -75.85 dollars.

When the computer is premium, the predicted price decreases with -399.39 dollars.

The adjusted R^2 is 0.504 which indicates there is a moderately strong correlation. Thus, 50.4% of the variance of the response variable can be explained by the explanatory variables, adjusted by the number of explanatory variables.

The factors of a computer that are associated with a high price according to our model are: no CD player, not premium, high speed, small hard disk, large RAM size, and a large screen.

The explanatory variable multi, which designates whether or not the computer has multiple ports, was not significant, which means that the model we made from the dataset does not provide sufficient evidence that there is an association between the price of a computer and whether or not the computer has multiple ports.

Our conclusions can probably not be generalised very well to future computer sales because the dataset that we used to fit the model contains a lot of datapoints of old computers. The model accurately predicts the price of computers like the ones in the dataset, but probably wouldn't do well for recent computers and ones that will be released in the future. For instance, in our model, there is a negative association between more harddrive space and price. In the past, computers were generally more expensive and also had smaller harddrives, so this might be where the association between a smaller harddrive and higher price comes from in our model.

However, today, a bigger harddrive is certainly associated with a higher price, because more recent programs need more space on the harddrive, and people use their computers to perform more tasks, which means they store more files on the computers. We confirmed that the negative association between harddrive space and price is due to entries of old computers by be refitting the model to just the last 359 data points (the further down you go in the dataset file, the more recent the computer is).

Second model:

$price = -135 + 3.72 * speed\_dummy2 + 0.22 * hd + 43.98 * ram\_dummy2 + 94.2 * screen\_dummy2 + 120.04 * cd\_dummy2 - 350.41 * premium\_dummy2$

For an increase of 1 in speed, the price increases with 3.72 dollars.

For an increase of 1 in hd, the price increases with 0.22 dollars.

For an increase of 1 in ram, the price increases with 43.98 dollars.

For an increase of 1 in screen, the price increases with 94.2 dollars.

When the computer has a cd player, the price increaes with 120.04 dollars.

When the computer is premium, the price decreases with -350.41 dollars.

The adjusted R^2 is 0.941 which indicates there is strong correlation. Thus, 94.1% of the variance of the response variable can be explained by the explanatory variables, adjusted by the number of explanatory variables.

Our new model shows a positive association between price and harddrive size and a positive association between price and cd_dummy, where these variables were negatively associated with price in our previous model.

1.6 Conclusion

We have now enough information to provide an answer to our initial research question. When we fitted the model, we found out that the explanatory variable "multi" did not provide any explanatory power. The other variables did seem to be associated with the price of a computer. If the alternative hypothesis was true, the variables speed, hd, ram, screen, cd, multi and premium would have to influence the price of the computer. This was not the case, we removed multi from our model because there was no significant evidence that, when all other variables held constant, this variable was associated with the price. Besides the variable multi, all other variables appeared to be associated with the price. The models are probably also not very well suited to predict the prices of computers in the future because it doesn't account for possible future developments. It doesn't seem inconceivable that better implementations of existing concepts will emerge in the future, which might replace existing components. You can also speculate that at some point, the performance (speed) of the computer will be negligible for most people because computers have become so fast that they can do anything they want to do with it, which might lead to other aspects of the computer becoming more important, such as design. This isn't accounted for in our model.

1.7 Limitations

Our dataset originally did not include the unit of measurements, thus we tried to find out which unit of measurements were most probable to be the one used in the dataset. We succeeded in doing this for most of the variables, although we were not able to find out the measurement unit of the ''speed" variable. We did not find out the meaning of the variables ''ads" and ''trends", which is why we did not consider them for either of the models. Furthermore, we expect the first model to no longer be valid due to the fact that it was modelled based on data from old computers. New computer parts are also developed in the computer industry at a very high rate and the newest computer parts are not taken into account in either of our model. To have a model which generalises over time, the model should be reassessed constantly to include changes in the computer industry.