Assignment 1 Report: Buying A Laptop From eBay

1 Introduction

This report aims to figure out the relationship between price and laptop specifics (processor speed, RAM space, hard drive space and type), selling status (sold or not) and selling strategy (auction or buy it now), and give suggestions on good deals, based on 220 records of laptop price data scraped from eBay. Particularly, to meet the customer's requirement and expectations, this report explains whether and how the hard drive type (solid state or not) and selling strategy (auction or buy it now) affect the price in detail, analyzes the prices of unsold buy-it-now laptops with a large hard drive and proposes the best price.

It is found that the prices of laptops that are buy-it-now, still not sold and have a solid-state hard drive and higher speed of processor are higher. The best price for an unsold laptop with a large hard drive is around $410.

2 Methods

Our goal of analysis can be divided into two parts: exploration on factors affecting laptop prices and recommendations on good deals.

2.1 Exploration on factors affecting prices

We conducted qualitative and quantitative analyses to explore the association between laptop prices and other factors.

Qualitatively, we calculated the average laptop prices and their 95% confidence intervals across different values for each factor, and then performed comparison by visualizing these average laptop prices and confidence intervals through bar-plots. For the two factors, hard drive type (solid state or not) and selling strategy (auction or buy it now), which customer is particularly interested in, we additionally performed T-test to examine whether there is a difference between the mean laptop prices of two groups (e.g. group 1: solid state, group 2: not solid state).

Quantitatively, since all prices are positive values and larger than some certain value, we constructed a Gamma regression model with log-link to fit the relationship between price and other factors. No obvious correlation between each two factors is detected, so we exclude interaction terms in our model. And then, based on this model, we explained whether a factor affects the price significantly by checking the P-value of the regression coefficient of this factor, and how a significant factor affects the price by observing the value and 95% confidence interval of the regression coefficient of this factor.

2.2 Recommendation on good deals

To give suggestion on good deals, we firstly found the lowest price satisfying the customer's requirements (large hard drive space and better to be buy-it-now) among all unsold laptops through filtering the without constructing models.

Furthermore, we fitted the Gamma regression model (selected by smallest mean squared error) using all data from those sold laptops and then used this model to predict prices for unsold laptops. We compared the difference between the predicted price and the real price given on eBay for each unsold laptop and we picked a cost-optimal laptop which maximizes the absolute value of the negative difference between the predicted price and real prices.

3 Results

3.1 Data Summary

The data we used are scraped from eBay and provided by the customer. The dataset contains 220 rows and 7 columns. Each row of the dataset is one listing on eBay, which records values of 7 variables. The details on definition, type and number of available values of these 7 variables are shown as Table 1.

Variable	Definition	Type	Number of Samples
`price`	Sale price, or listing price if it isn't sold.	Numerical	non-NA - 220
`sale`	Whether the listing sold or not.	Categorical	SOLD - 158; NOT SOLD - 63.
`ghz`	Speed of the processor in GHz.	Numerical	non-NA - 171; NA - 49.
`ram`	Amount of ram in GB.	Numerical	non-NA - 177; NA - 43.
`hd`	Amount of hard drive space, in GB. (for both magnetic and ssd drives)	Numerical	non-NA - 150; NA - 70.
`ssd`	Whether the hard drive is magnetic (No) or solid state (SSD).	Categorical	SSD - 80; No - 140.
`BIN`	whether it was an auction (FALSE) or buy it now (TRUE)	Categorical	TRUE - 119; FALSE - 101.

Table 1: Detailed information on variables.

It can be noticed that there are large amount of NA's in the dataset and missing values only exist for three numerical variables, `ghz`, `ram` and `hd`, which are all numerical. There are 92 rows (listings) that contain at least one NA value, which is around 42% of total number of rows. Removing these samples containing NAs would result in non-negligible information loss. Here, we imputed the missing value of a listing with mean of its 3 nearest neighbors' values (nearest neighbor of a listing containing NA values is one listing in all listings not containing NA values, which gives the largest similarity to the listing with NA values among all listings without NA values).

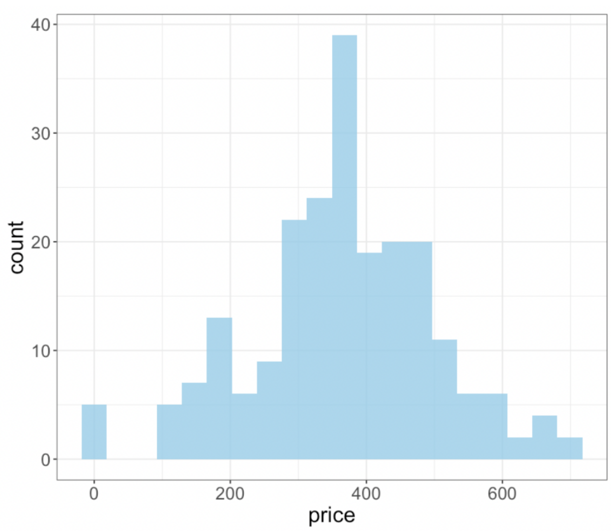


Figure 1: Histograms for `price`.

As Figure 1 shows and with further check in the dataset, 5 of the `price` values are pretty small, which is not larger than 10. Such low price for a laptop even for a pre-owned laptop is not realistic. 4 out of 5 such laptops are not sold. These listings are more likely to be fake, so we didn't consider these laptops.

After imputation for missing values and removing outliers, the distribution metrics for all variables are shown in Table 2.

Variable	Mean	Standard Deviation	Median	IQR
Sale/Listing Price in USD	372.59	8.31	361.00	150.00
Processor Speed in GHz	2.56	0.01	2.50	0.10
RAM in GB	5.72	0.14	5.00	4.00
Hard Drive Space in GB	218.16	6.81	192.00	175.50
Sold or Not	SOLD: 157; NOT SOLD: 58			
Solid State (SSD) or Magnetic(No) Hard Drive	SSD: 80; No: 135			
Buy-it-now (TRUE) or Auction (FALSE)	TRUE: 114; FALSE: 101			

Table 2: Distribution metrics.

3.2 Exploration on factors affecting prices

Qualitatively, prices are higher for laptops that have been sold, are buy-it-now, and have solid state hard drive, higher speed of processor, 4GB or 8GB RAM space and around 120GB or 300GB amount of hard drive space.

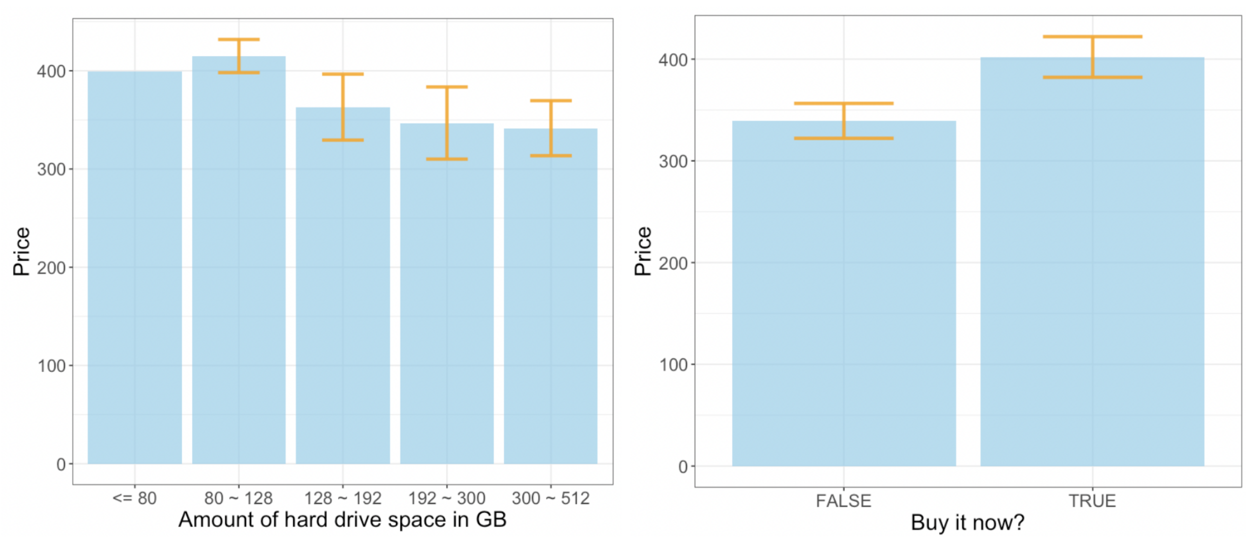


Figure 2: Average prices for different values for `hd` (left) and `BIN` (right) (orange bar stands for confidence interval).

Figure 2 shows the bar plots of average prices for different value for `hd` and `BIN`, two variables of customer's interest. For `hd`, we can see that the group $80 - 128$ (larger than 80GB and not larger than 128GB) gives the highest price. Some laptops with small hard drive space have higher price. And for `BIN`, buy-it-now laptops have higher price than laptops at auction. The t-test for the buy-it-now and auction laptops gives a P value smaller than 0.01, which suggests we can say the mean price for buy-it-now and auction laptops are different at the significance level of 0.01.

Quantitatively, the coefficients given by Gamma regression model are shown in Table 3.

Variable	Rate Ratio (CI)	P-value
Processor Speed in GHz	1.77 (1.21, 2.60)	< 0.01
RAM in GB	1.01 (0.99, 1.03)	0.24
Hard Drive Space in GB	1.00 (1.00, 1.00)	0.24
Sold vs Not	0.89 (0.82, 0.97)	0.01
Solid State vs Magnetic Hard Drive	1.34 (1.21, 1.48)	< 0.01
Buy-it-now vs Auction	1.19 (1.10, 1.29)	< 0.01

Table 3: Regression parameters and p-values for laptop prices.

The p-values of RAM space (`ram`) and hard drive space (`hd`) are 0.24 (larger than significance level 0.05). Based on the model, it indicates that the RAM space and hard drive space would not affect price significantly at the level of 0.05. The other four variables, processor speed (`ghz`), sold or not (`sale`), solid state hard drive or not (`ssd`) and buy-it-now or not (`BIN`), are significant at the level of 0.05. The laptop price would increase by 177% (121%, 260%) if the processor speed increase by 1 GHz; The expected price of a sold laptop are 89% (82%, 97%) of the expected price of an unsold laptop; The price of a laptop with solid state hard drive would be 34% (21%, 48%) higher than the price of a laptop with magnetic hard drive; The price of a buy-it-now laptop with solid state hard drive would be 19% (10%, 29%) higher than the price of a laptop at auction.

3.3 Recommendation on good deals

The customer prefers large hard drive spac, solid state drive and a buy-it-now laptop if buy-it-now of auction does not affect the price. We know that the price of a buy-it-now laptop with solid state hard drive is higher than the price of a laptop at auction from regression model in 3.2, so the buy-it-now preference is relaxed.

We firstly filter the lowest price satisfying the preference among not-yet sold laptops. The median of hard drive space is 192GB (Table 2). If we set 192GB for hard drive space and solid state drive as the criteria, there would only be one unsold listing satisfying the requirements (Table 4)

Variable	Value
Listing Price in USD	565
Processor Speed in GHz	2.5
RAM in GB	8
Hard Drive Space in GB	240
Sold vs Not	Not Sold
Solid State vs Magnetic Hard Drive	Solid State
Buy-it-now vs Auction	Buy-it-now

Table 4: Listing satisfying solid state drive and $>$ 192GB hard drive space.

The listing price may not be reasonable enough, which is much higher than the median of all sale/listing price. If we relax the hard drive space from 192GB to 128GB (25% quantile of all hard drive spaces), then we'll more choices. The lowest 3 prices are shown in Table 5. The first two are in the same condition and the price is reasonable (slightly higher than the median of all sale/listing price) and they are both at auction. If buy-it-now is still preferred, the third listing would be a good choice, a little higher price with higher processor speed and buy-it-now selling strategy.

Variable	Listing 1	Listing 2	Listing 3
Listing Price in USD	373	373	410
Processor Speed in GHz	2.5	2.5	2.6
RAM in GB	8	8	8
Hard Drive Space in GB	128	128	128
Sold vs Not	Not Sold	Not Sold	Not Sold
Solid State vs Magnetic Hard Drive	Solid State	Solid State	Solid State
Buy-it-now vs Auction	Auction	Auction	Buy-it-now

Table 5: Listings satisfying solid state drive and > 128GB hard drive space (lowest 3 listing prices).

Apart from the lowest, we used our Gamma regression model to predict all prices for unsold laptops and compared the difference between predicted price and listing price for each laptop. The most cost-optimal 3 listings are shown in Table 6.

Variable	Listing 1	Listing 2	Listing 3
Difference between Predicted Price and Listing Price	-66.62	-15.97	-15.97
Predicted Price in USD	476.62	388.97	388.97
Listing Price in USD	410.00	373.00	373.00
Processor Speed in GHz	2.6	2.5	2.5
RAM in GB	8	8	8
Hard Drive Space in GB	128	128	128
Sold vs Not	Not Sold	Not Sold	Not Sold
Solid State vs Magnetic Hard Drive	Solid State	Solid State	Solid State
Buy-it-now vs Auction	Buy-it-now	Auction	Auction

Table 6: Listings satisfying solid state drive and > 128GB hard drive space (most cost-optimal 3 listings).

It can be seen that the most cost-optimal 3 listing in Table 6 are the same as 3 listings with lowest prices in Table 5. Based on the difference between predicted price and listing price, listing 1 is the optimal deal.

4 Conclusion

From this work, we analyzed the relationship between laptop price (`price`) and 6 variables including processor speed (`ghz`), RAM space (`ram`), hard drive space (`hd`), sale status (`sale`), hard drive type (`ssd`) and selling strategy (`BIN`) qualitatively by calculating and visualizing averaged prices for difference values and quantitatively by constructing a Gamma regression model to help the customer understand what factors or variables would affect and how they would affect the laptop price. And we also recommended on the best deal that gives reasonable low price and meets the customer's preference: large hard drive space, solid state drive and better to be buy-it-now if it doesn't affect the price.

We found from the bar plots of averaged prices for difference values that prices are higher for laptops that have been sold, are buy-it-now, and have solid state hard drive, higher speed of processor, 4GB or 8GB RAM space and around 120GB or 300GB amount of hard drive space. Furthermore, based on the Gamma regression model fitting results, only the processor speed (`ghz`), sale status (`sale`), hard drive type (`ssd`) and selling strategy (`BIN`) are significant. Keeping the other variables not changing, the laptop price would increase by 177% (121%, 260%) if the processor speed increase by 1 GHz; the expected price of a sold laptop are 89% (82%, 97%) of the expected price of an unsold laptop; the price of a laptop with solid state hard drive would be 34% (21%, 48%) higher than the price of a laptop with magnetic hard drive; the price of a buy-it-now laptop with solid state hard drive would be 19% (10%, 29%) higher than the price of a laptop at auction.

There are several limitations in this work. Firstly, in this work, around 42% of data has missing values. Although we imputed them with nearest neighbors' values, our analysis may be biased and not accurate enough. Secondly, the number of samples is limited (only 220 listings), which made our conclusions probably not be general enough. Our recommendations on good deals were only based on these 220 listings and they may be not optimal if more listings are considered.

Appendix

All analyses and codes with comments can be found at STATS504_Assignment1_Appendix.pdf (supports preview).

You can also view from next page to see the attached version.

Assignment 1: Buying a laptop from eBay

In [1]:
```
# load required library
options(warn = -1)
require(tidyverse)
require(ggplot2)
require(GGally)
require(lmtest)
require(randomForest)
```

```
Loading required package: tidyverse

── Attaching packages ───────────────────────────────────────
──────────────────────────────────────────── tidyverse 1.3.1 ──

✔ ggplot2 3.3.6     ✔ purrr   0.3.4
✔ tibble  3.1.7     ✔ dplyr   1.0.9
✔ tidyr   1.2.0     ✔ stringr 1.4.0
✔ readr   2.1.2     ✔ forcats 0.5.1

── Conflicts ────────────────────────────────────────────────
────────────────────────────────────── tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()

Loading required package: GGally

Registered S3 method overwritten by 'GGally':
  method from
  +.gg   ggplot2

Loading required package: lmtest

Loading required package: zoo


Attaching package: ‘zoo’


The following objects are masked from ‘package:base’:

    as.Date, as.Date.numeric


Loading required package: randomForest

randomForest 4.7-1.1

Type rfNews() to see new features/changes/bug fixes.


Attaching package: ‘randomForest’


The following object is masked from ‘package:dplyr’:

    combine


The following object is masked from ‘package:ggplot2’:

    margin
```

1. Data Load and Preprocessing

In [2]:
```
# load data
laptop = read.csv("laptopData.csv", row.names = 1)
head(laptop)
```

A data.frame: 6 × 7

	sale	price	ghz	ram	hd	ssd	BIN
	<chr>	<dbl>	<dbl>	<int>	<int>	<chr>	<lgl>
1	SOLD	404.99	2.7	8	NA	SSD	FALSE
2	SOLD	355.00	2.5	8	128	SSD	FALSE
3	SOLD	449.99	2.6	4	128	No	TRUE
4	NOT SOLD	499.99	2.5	4	320	No	TRUE
5	NOT SOLD	199.99	NA	NA	NA	No	TRUE
6	NOT SOLD	699.95	2.5	4	128	SSD	TRUE

In [3]:
```
# the number of total samples
nrow(laptop)
```

220

In [4]:
```
# change sale, ssd and BIN to factor
laptop$sale = as.factor(laptop$sale)
laptop$ssd = as.factor(laptop$ssd)
laptop$BIN = as.factor(laptop$BIN)
```

In [5]:
```
head(laptop)
```

A data.frame: 6 × 7

	sale	price	ghz	ram	hd	ssd	BIN
	<fct>	<dbl>	<dbl>	<int>	<int>	<fct>	<fct>
1	SOLD	404.99	2.7	8	NA	SSD	FALSE
2	SOLD	355.00	2.5	8	128	SSD	FALSE
3	SOLD	449.99	2.6	4	128	No	TRUE
4	NOT SOLD	499.99	2.5	4	320	No	TRUE
5	NOT SOLD	199.99	NA	NA	NA	No	TRUE
6	NOT SOLD	699.95	2.5	4	128	SSD	TRUE

1.1 Basic Statistics

In [6]:
```
summary(laptop)
```

```
       sale          price             ghz              ram
 NOT SOLD: 62   Min.   :  1.0   Min.   :2.500   Min.   : 2.000
 SOLD    :158   1st Qu.:300.0   1st Qu.:2.500   1st Qu.: 4.000
                Median :357.5   Median :2.500   Median : 4.000
                Mean   :364.2   Mean   :2.573   Mean   : 5.814
                3rd Qu.:450.0   3rd Qu.:2.600   3rd Qu.: 8.000
                Max.   :700.0   Max.   :3.200   Max.   :16.000
                                NA's   :49      NA's   :43
       hd           ssd         BIN
 Min.   : 80.0   No :140   FALSE:101
 1st Qu.:128.0   SSD: 80   TRUE :119
 Median :160.0
 Mean   :205.2
 3rd Qu.:300.0
 Max.   :512.0
 NA's   :70
```

1.2 Missing Values

In [7]:
```
# find all rows that contain NAs
NAs = laptop[rowSums(is.na(laptop)) > 0, ]
nrow(NAs)
```

92

In [8]:
```
colnames(laptop)[colSums(is.na(laptop)) > 0]
```

'ghz' · 'ram' · 'hd'

There are 92 rows with NAs. However, we only have 220 samples in total. Removing these samples containing NAs would result in non-negligible information loss. We need to find some way to impute the missing values. Missing values only exist for three variables, `ghz` , `ram` and `hd` , which are all numerical. Here, I'll impute the missing value of a sample with mean of its 3 nearest neighbors' value.

In [9]:
```
test = colnames(NAs)[!is.na(NAs[1,])]
names(test) = c('x1', 'x2', 'x3', 'x4', 'x5', 'x6')
```

In [10]:
```
test[c('x1', 'x2')]
```

x1: 'sale' **x2:** 'price'

In [11]:
```
# calculate the distance between samples with missing values and samples wit
non_NAs = laptop[rowSums(is.na(laptop)) == 0,]
vars = colnames(NAs)
col_range = as.numeric(sapply(non_NAs[, c('price', 'ghz', 'ram', 'hd')], fur
names(col_range) = c('price', 'ghz', 'ram', 'hd')
dist = matrix(0, nrow = nrow(NAs), ncol = nrow(non_NAs))
for (i in 1:nrow(dist)) {
    NA_sample = NAs[i, ]
    var = vars[!is.na(NA_sample)]
    cat_var = var[var %in% c('sale', 'ssd', 'BIN')]
    num_var = var[var %in% c('price', 'ghz', 'ram', 'hd')]
    for (j in 1:ncol(dist)) {
        num_part = abs(NAs[i, num_var]-non_NAs[j, num_var])/col_range[num_va
        cat_part = (NAs[i, 'sale']!=non_NAs[j, 'sale'])+(NAs[i, 'ssd']!=non_
        dist[i,j] = (sum(num_part)+sum(cat_part))/(length(num_var)+3)
    }
}
```

In [12]:
```
# impute the missing value with 3 nearest neighbors
for (i in 1:nrow(dist)) {
    idx = order(dist[i,])[1:3]
    NA_sample = NAs[i, ]
    var = vars[is.na(NA_sample)]
    if (length(var) == 1) NAs[i, var] = mean(non_NAs[idx, var])
    else NAs[i, var] = colMeans(non_NAs[idx, var])
}
```

In [13]:
```
# round ram and hd to int, ghz to 1st decimal to keep consistent to original
NAs$ram = round(NAs$ram, 0)
NAs$hd = round(NAs$hd, 0)
NAs$ghz = round(NAs$ghz, 1)
```

In [14]:
```
head(NAs)
```

A data.frame: 6 × 7

	sale	price	ghz	ram	hd	ssd	BIN
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<fct>	<fct>
1	SOLD	404.99	2.7	8	160	SSD	FALSE
5	NOT SOLD	199.99	2.5	5	192	No	TRUE
9	SOLD	128.00	2.7	6	267	No	FALSE
16	SOLD	600.00	2.6	4	192	No	TRUE
17	SOLD	203.52	2.5	7	380	No	FALSE
19	SOLD	109.19	2.5	4	253	No	TRUE

In [15]:
```
# concatenate NAs and non_NAs to form new data set
laptop_new = rbind(NAs, non_NAs)[order(as.numeric(c(rownames(NAs), rownames(
```

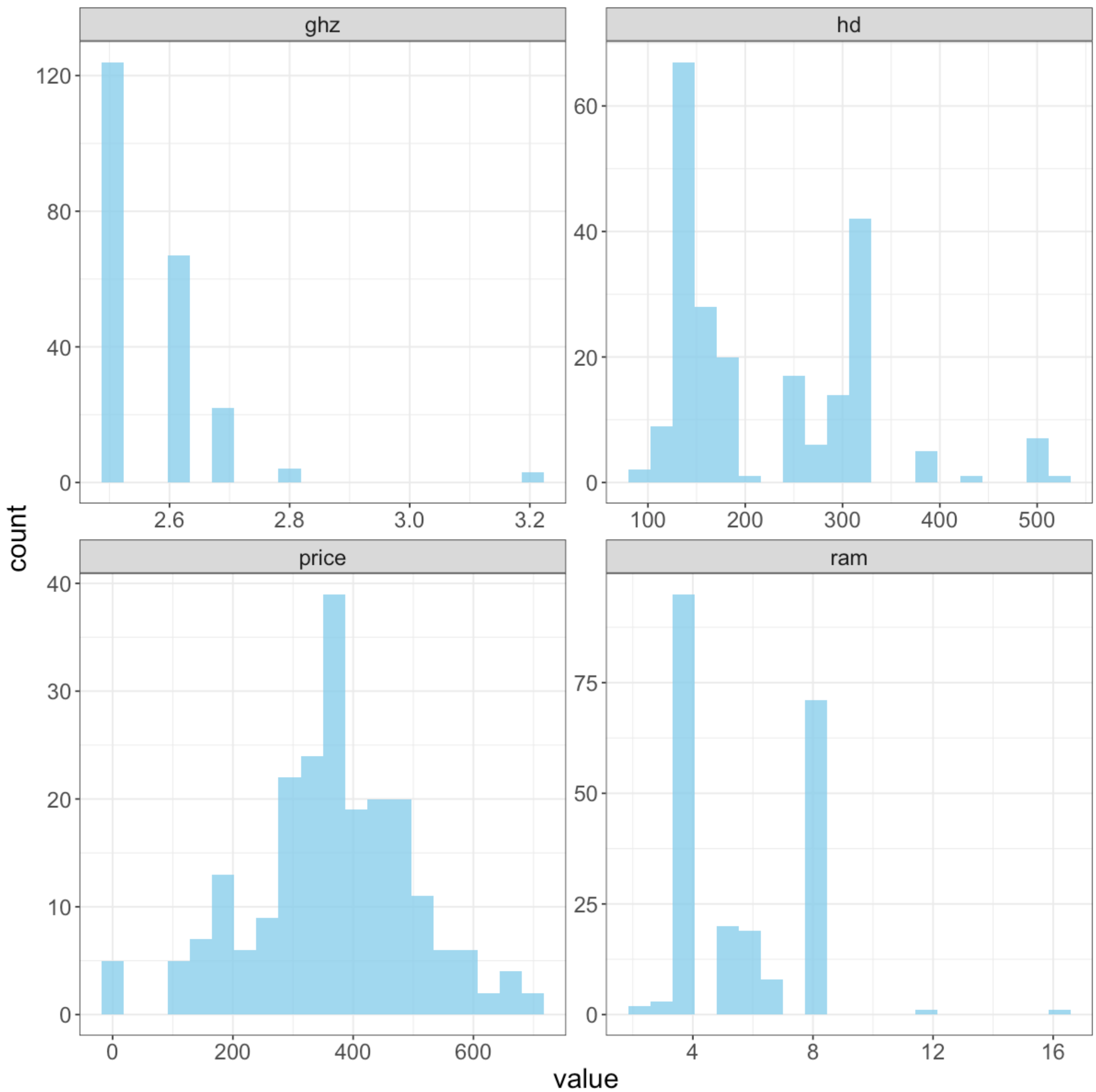
In [16]:
```
anyNA.data.frame(laptop_new) # no NAs in this data set
```

FALSE

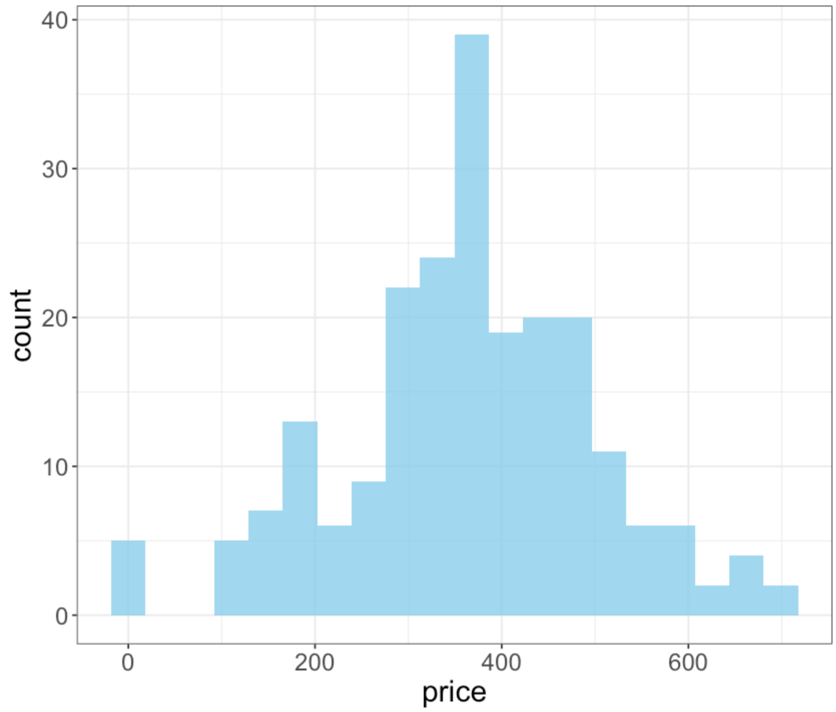
1.3 Basic Plots

In [17]:
```
laptop.hist = laptop_new[,sapply(laptop_new, is.numeric)] # filter all numer
laptop.bar = laptop_new[,sapply(laptop_new, is.factor)] # filter all factor
# melt the dataframe to plot
laptop.hist = laptop.hist %>% gather(key = "variable", value = "value")
laptop.bar = laptop.bar %>% gather(key = "variable", value = "value")
```

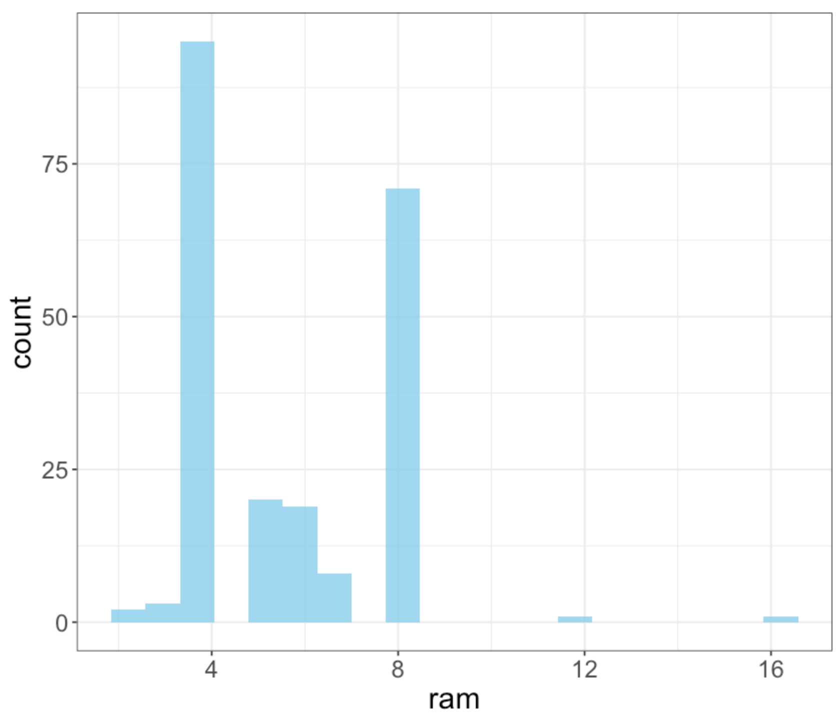
In [18]:
```
options(repr.plot.width = 10, repr.plot.height = 10)
# histogram for numerical variables
laptop.hist %>% ggplot() +
    geom_histogram(aes(x = value), bins = 20, fill="skyblue", alpha=0.8) +
    facet_wrap(~variable, scales = 'free') + theme_bw() +
    theme(text = element_text(size = 18))
```



In [19]:
```
options(repr.plot.width = 7, repr.plot.height = 6)
laptop_new %>% ggplot() +
    geom_histogram(aes(x = price), bins = 20, fill="skyblue", alpha=0.8) + t
    theme(text = element_text(size = 18))
```



In [20]:
```
options(repr.plot.width = 7, repr.plot.height = 6)
laptop_new %>% ggplot() +
    geom_histogram(aes(x = ram), bins = 20, fill="skyblue", alpha=0.8) + the
    theme(text = element_text(size = 18))
```



- It can be seen that some of the price values are pretty small, which is not larger than 10. Such low price for a laptop even for a pre-owned laptop is not realistic and we can see 4 out of 5 such laptops are not sold. These posts are more likely to be fake, so we'll not consider these laptops in our model.
- Since all prices are larger than some value, we may use a Gamma regression model or log-normal regression model to fit it.
- Most of laptops on the market has
  - 2.5Ghz or 2.6Ghz speed of processor,
  - around 150GB or 320GB hard drive,
  - 4GB or 8GB ram.

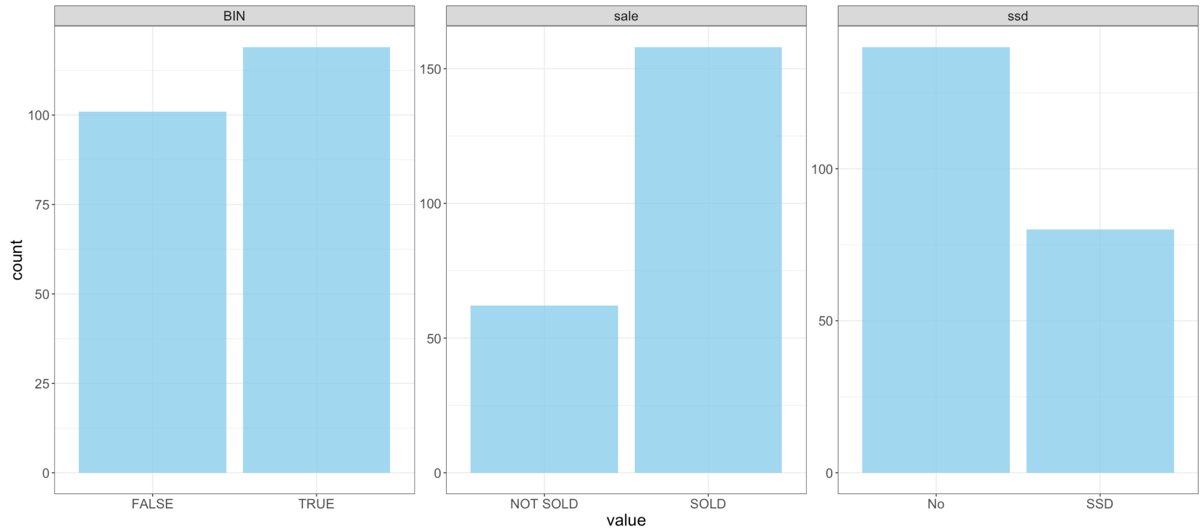
In [21]:
```
subset(laptop_new, price < 50)
```

A data.frame: 5 × 7

	sale	price	ghz	ram	hd	ssd	BIN
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<fct>	<fct>
73	NOT SOLD	1	2.5	4	320	No	TRUE
74	NOT SOLD	4	2.5	4	320	No	TRUE
84	NOT SOLD	1	2.8	8	160	No	TRUE
87	NOT SOLD	1	2.8	8	160	No	TRUE
203	SOLD	1	2.5	4	253	No	TRUE

In [22]:
```
laptop_new = subset(laptop_new, price > 50)
```

In [23]:
```
options(repr.plot.width = 18, repr.plot.height = 8)
# barplot for categorical variables
laptop.bar %>% ggplot() +
    geom_bar(aes(x = value), fill="skyblue", alpha=0.8) + facet_wrap(~variab
    theme(text = element_text(size = 18))
```



Everything else looks normal to me.

In [24]:
```
summary(laptop_new)
```

```
       sale          price             ghz               ram        
 NOT SOLD: 58   Min.   :103.8   Min.   :2.500   Min.   : 2.000  
 SOLD    :157   1st Qu.:300.0   1st Qu.:2.500   1st Qu.: 4.000  
                Median :361.0   Median :2.500   Median : 5.000  
                Mean   :372.6   Mean   :2.564   Mean   : 5.726  
                3rd Qu.:450.0   3rd Qu.:2.600   3rd Qu.: 8.000  
                Max.   :700.0   Max.   :3.200   Max.   :16.000  
       hd          ssd          BIN         
 Min.   : 80.0   No :135   FALSE:101  
 1st Qu.:128.0   SSD: 80   TRUE :114  
 Median :192.0                        
 Mean   :218.2                        
 3rd Qu.:303.5                        
 Max.   :512.0                        
```

In [25]:
```
# mean
apply(laptop_new[c('price', 'ghz', 'ram', 'hd')], 2, mean)
```

price: 372.594418604651 **ghz:** 2.56418604651163 **ram:** 5.72558139534884 **hd:** 218.162790697674

In [26]:
```
# standard deviation
apply(laptop_new[c('price', 'ghz', 'ram', 'hd')], 2, function(x){sqrt(var(x)
```

price: 8.30771483223511 **ghz:** 0.00710869246505618 **ram:** 0.136302035478774 **hd:** 6.80909159315638

In [27]:
```
# median
apply(laptop_new[c('price', 'ghz', 'ram', 'hd')], 2, median)
```

price: 361 **ghz:** 2.5 **ram:** 5 **hd:** 192

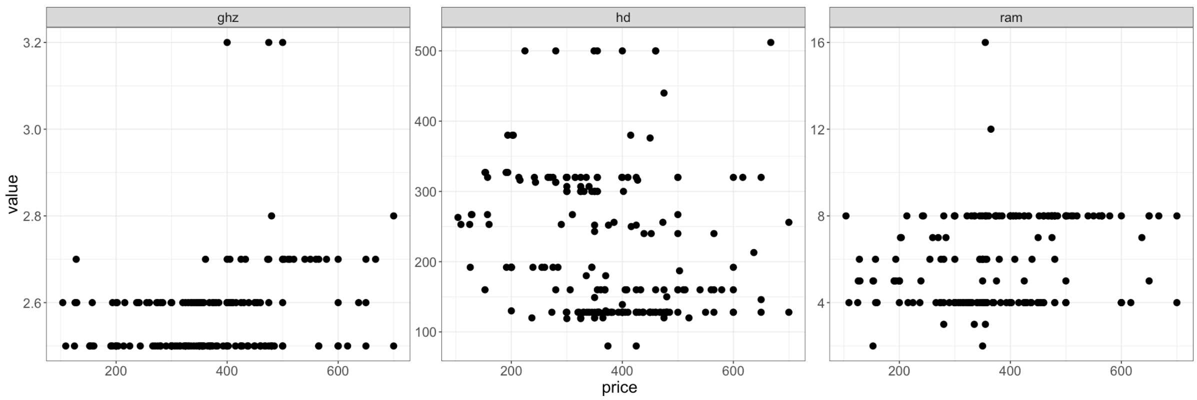
In [28]:
```
# IQR
apply(laptop_new[c('price', 'ghz', 'ram', 'hd')], 2, function(x){quantile(x,
```

price: 150 **ghz:** 0.1 **ram:** 4 **hd:** 175.5

2. Exploratory Data Analysis (EDA)

2.1 Numerical Variables

In [29]:
```
options(repr.plot.width = 18, repr.plot.height = 6)
laptop_new[,c('price', 'ghz', 'hd', 'ram')] %>% gather(key = 'variable', val
    ggplot() +
    geom_point(aes(x = price, y = value), size = 3) +
    facet_wrap(~variable, scales = 'free') + theme_bw() +
    theme(text = element_text(size = 18))
```



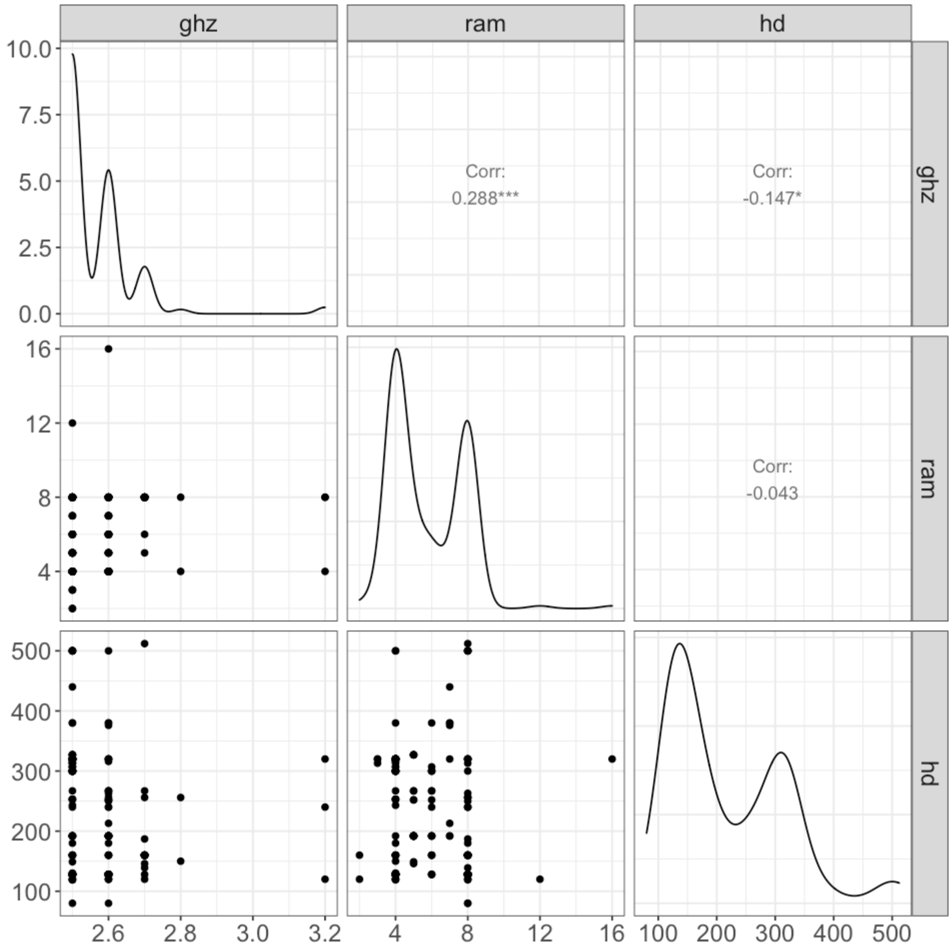
- No special trend is found. It seems that price has no obvious relationship with `ghz` , `hd` and `ram` .
- 16 and 12 seem to be outliers for `ram` ; 3.2 seem to be outlier for `ghz` .

Pairwsie Correlation

In [30]:
```
colnames(laptop_new)
```

'sale' · 'price' · 'ghz' · 'ram' · 'hd' · 'ssd' · 'BIN'

In [31]:
```
options(repr.plot.width=8, repr.plot.height=8)
laptop_new %>% ggpairs(columns = c(3, 4, 5)) + theme_bw() +
    theme(text = element_text(size = 18))
```

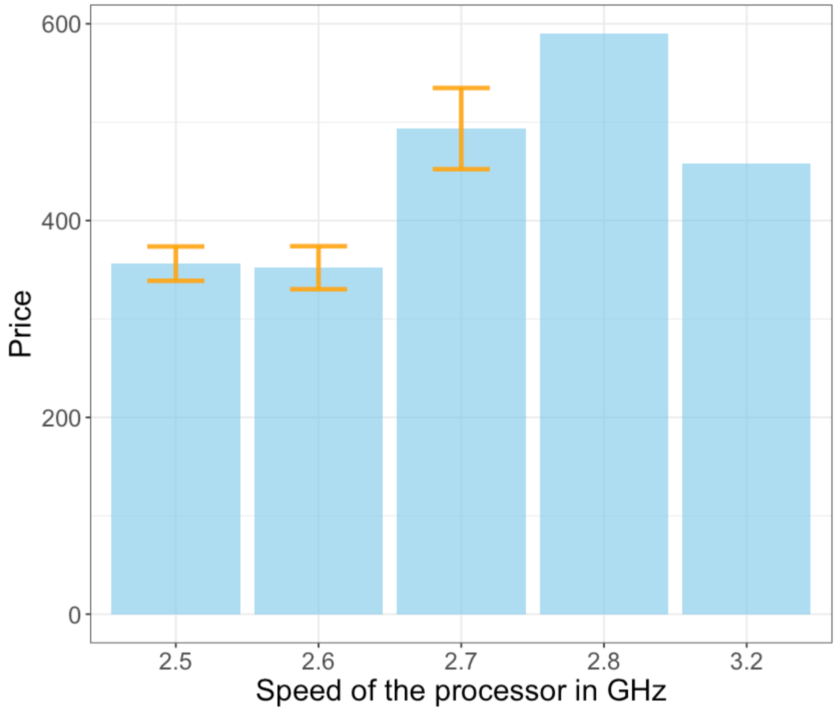


No obvious correlation is found.

Does price vary with ghz?

In [32]:

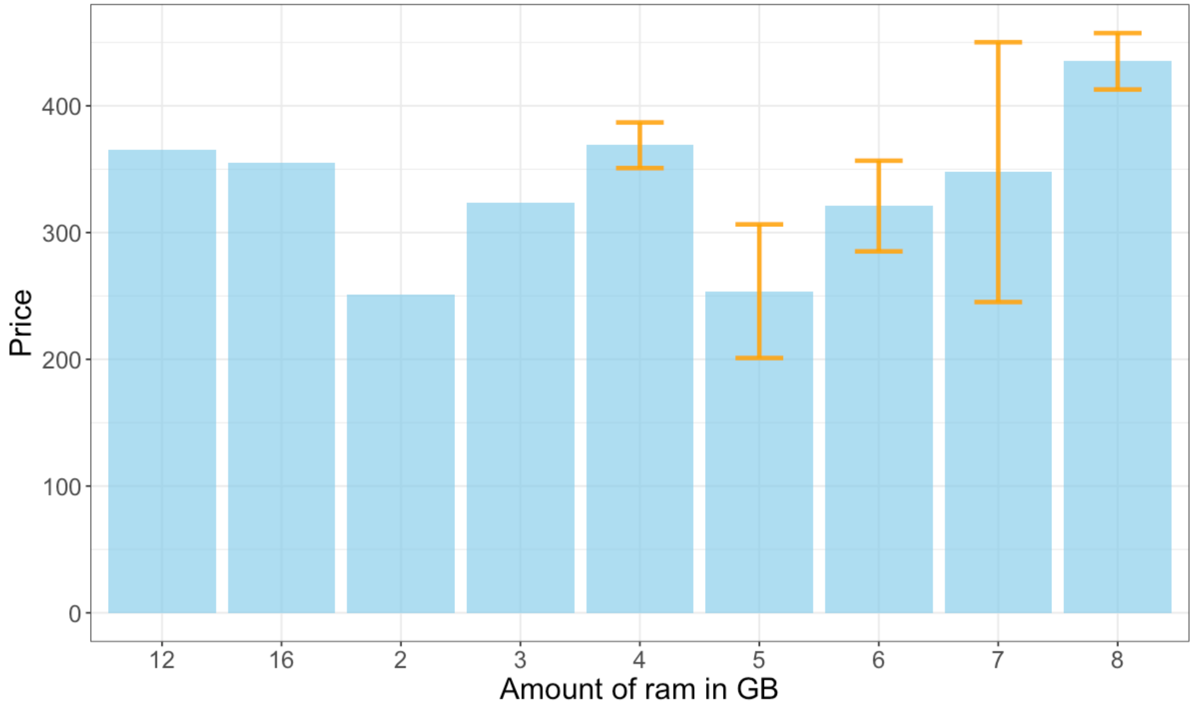
```
options(repr.plot.width = 7, repr.plot.height = 6)
price_ghz = laptop_new %>% group_by(ghz) %>%
    summarize(mean_price = mean(price), count = n(), std_price = qt(0.95, n(
price_ghz$std_price[which(price_ghz$count <= 5)] = NA
price_ghz %>% ggplot() +
    geom_bar(aes(x=as.character(ghz), y=mean_price),
             stat="identity", fill="skyblue", alpha=0.7) +
    geom_errorbar(aes(x=as.character(ghz), ymin=mean_price-std_price, ymax=m
                  width=0.4, colour="orange", alpha=0.9, size=1.3) +
    theme_bw() + theme(text = element_text(size = 18)) +
    xlab("Speed of the processor in GHz") + ylab("Price")
```



Does price vary with ram?

In [33]:

```
options(repr.plot.width = 10, repr.plot.height = 6)
price_ram = laptop_new %>% group_by(ram) %>%
    summarize(mean_price = mean(price), count = n(), std_price = qt(0.95, n(
price_ram$std_price[which(price_ram$count <= 5)] = NA
price_ram %>% ggplot() +
    geom_bar(aes(x=as.character(ram), y=mean_price),
             stat="identity", fill="skyblue", alpha=0.7) +
    geom_errorbar(aes(x=as.character(ram), ymin=mean_price-std_price, ymax=m
                  width=0.4, colour="orange", alpha=0.9, size=1.3) +
    theme_bw() + theme(text = element_text(size = 18)) +
    xlab("Amount of ram in GB") + ylab("Price")
```



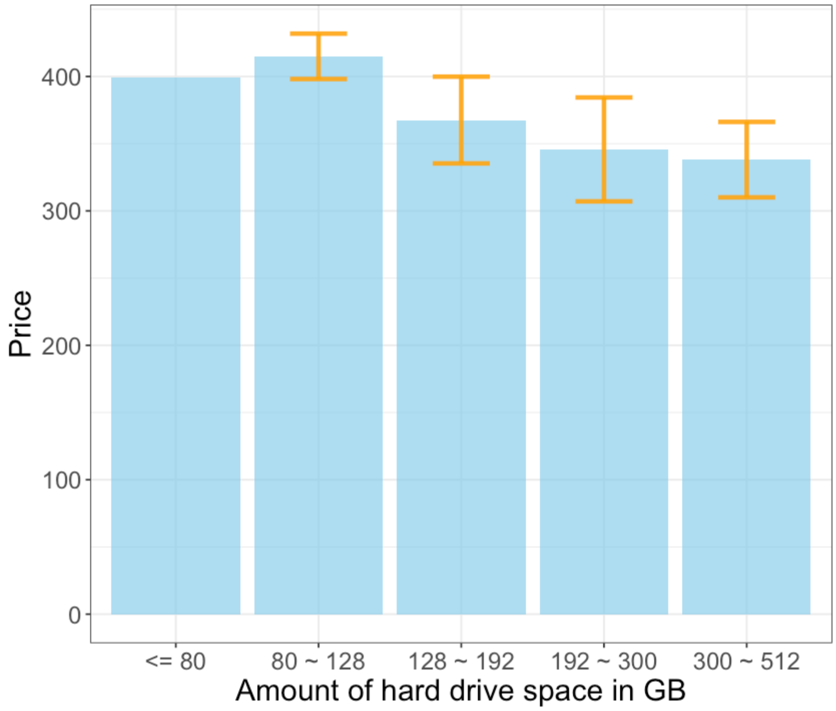
Does price vary with hd?

In [34]:
```
quantile(laptop_new$hd)
```

0%: 80 **25%:** 128 **50%:** 192 **75%:** 303.5 **100%:** 512

In [35]:
```
laptop_new$hdc = case_when(laptop_new$hd <= 80 ~ '<= 80',
                           laptop_new$hd > 80 & laptop_new$hd <= 128 ~ '80 ~
                           laptop_new$hd > 128 & laptop_new$hd <= 192 ~ '128
                           laptop_new$hd > 192 & laptop_new$hd <= 300 ~ '192
                           laptop_new$hd > 300 & laptop_new$hd <= 512 ~ '300
                          )
laptop_new$hdc = factor(laptop_new$hdc, level = c('<= 80', '80 ~ 128', '128
```

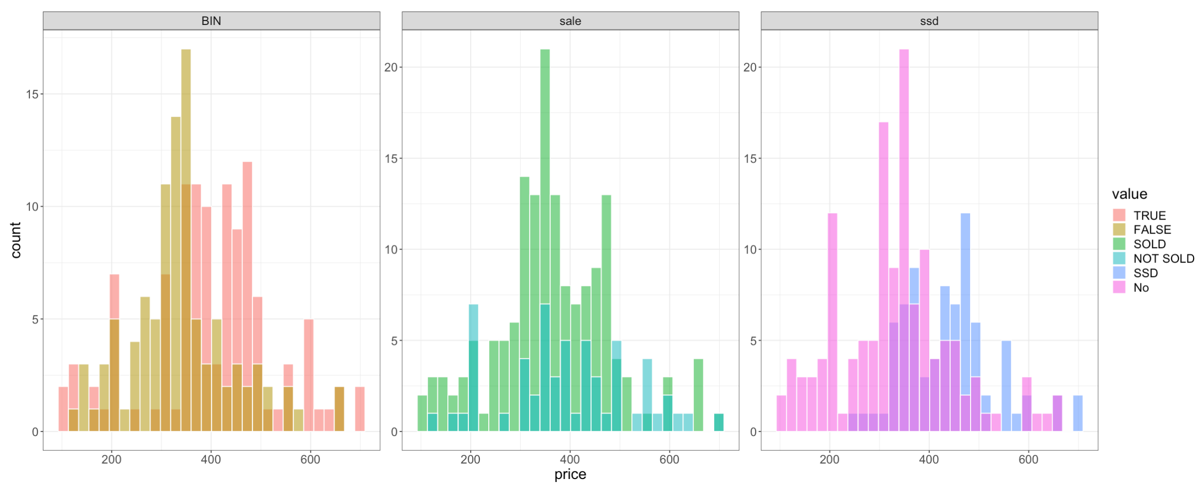
In [36]:
```
options(repr.plot.width = 7, repr.plot.height = 6)
price_hd = laptop_new %>% group_by(hdc) %>%
    summarize(mean_price = mean(price), count = n(), std_price = qt(0.95, n(
price_hd$std_price[which(price_hd$count <= 5)] = NA
price_hd %>% ggplot() +
    geom_bar(aes(x=hdc, y=mean_price),
             stat="identity", fill="skyblue", alpha=0.7) +
    geom_errorbar(aes(x=hdc, ymin=mean_price-std_price, ymax=mean_price+std_
                  width=0.4, colour="orange", alpha=0.9, size=1.3) +
    theme_bw() + theme(text = element_text(size = 18)) +
    xlab("Amount of hard drive space in GB") + ylab("Price")
```



2.2 Categorical Variables

In [37]:

```
options(repr.plot.width = 20, repr.plot.height = 8)
cat.df = laptop_new[,c('price', 'sale', 'ssd', 'BIN')] %>% gather(key = 'var
cat.df$value = factor(cat.df$value, levels = c('TRUE', 'FALSE', 'SOLD', 'NOT
cat.df %>% ggplot() +
    geom_histogram(mapping = aes(x = price, fill = value),
                   alpha = 0.6, bins = 30, position = "identity", color = "w
    facet_wrap(~variable, scales = 'free') + theme_bw() +
    theme(text = element_text(size = 18))
```

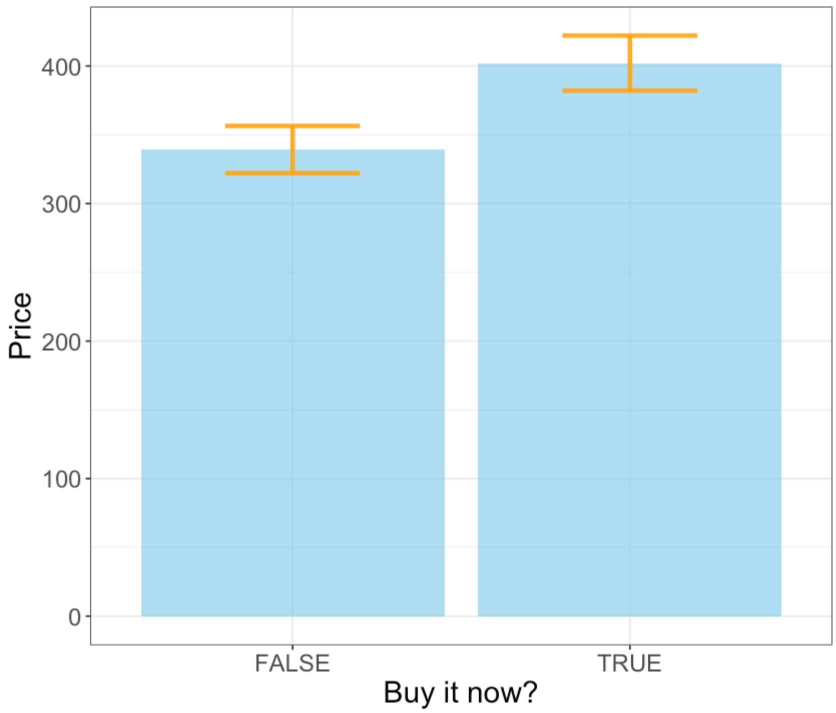


Roughly speaking,

- price is higher for laptop which we can buy now (BIN TRUE);
- price is higher for laptop with solid-state drive (SSD).

Does price vary with BIN?

In [38]:
```
options(repr.plot.width = 7, repr.plot.height = 6)
price_BIN = laptop_new %>% group_by(BIN) %>%
    summarize(mean_price = mean(price), count = n(), std_price = qt(0.95, n(
price_BIN$std_price[which(price_BIN$count <= 5)] = NA
price_BIN %>% ggplot() +
    geom_bar(aes(x=BIN, y=mean_price),
             stat="identity", fill="skyblue", alpha=0.7) +
    geom_errorbar(aes(x=BIN, ymin=mean_price-std_price, ymax=mean_price+std_
                  width=0.4, colour="orange", alpha=0.9, size=1.3) +
    theme_bw() + theme(text = element_text(size = 18)) +
    xlab("Buy it now?") + ylab("Price")
```



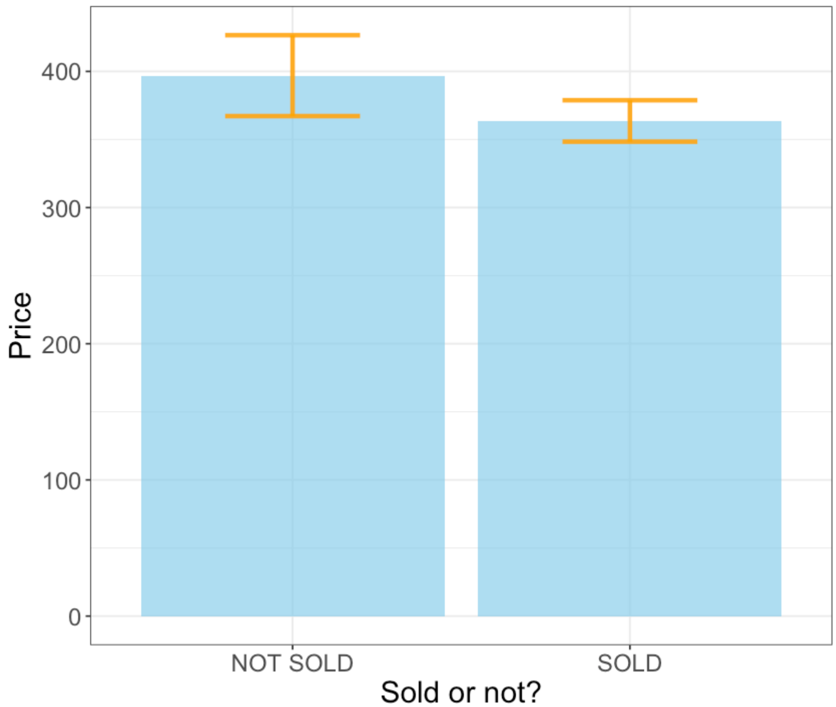
In [39]:
```
t.test(laptop_new$price[which(laptop_new$BIN == 'TRUE')], laptop_new$price[w
```

```
	Welch Two Sample t-test

data:  laptop_new$price[which(laptop_new$BIN == "TRUE")] and laptop_new$pri
ce[which(laptop_new$BIN == "FALSE")]
t = 3.9565, df = 211.24, p-value = 0.0001038
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 31.51742 94.10750
sample estimates:
mean of x mean of y 
 402.1017  339.2892 
```

Does price vary with sale?

In [40]:
```
options(repr.plot.width = 7, repr.plot.height = 6)
price_sale = laptop_new %>% group_by(sale) %>%
    summarize(mean_price = mean(price), count = n(), std_price = qt(0.95, n(
price_sale$std_price[which(price_sale$count <= 5)] = NA
price_sale %>% ggplot() +
    geom_bar(aes(x=sale, y=mean_price),
             stat="identity", fill="skyblue", alpha=0.7) +
    geom_errorbar(aes(x=sale, ymin=mean_price-std_price, ymax=mean_price+std
                  width=0.4, colour="orange", alpha=0.9, size=1.3) +
    theme_bw() + theme(text = element_text(size = 18)) +
    xlab("Sold or not?") + ylab("Price")
```



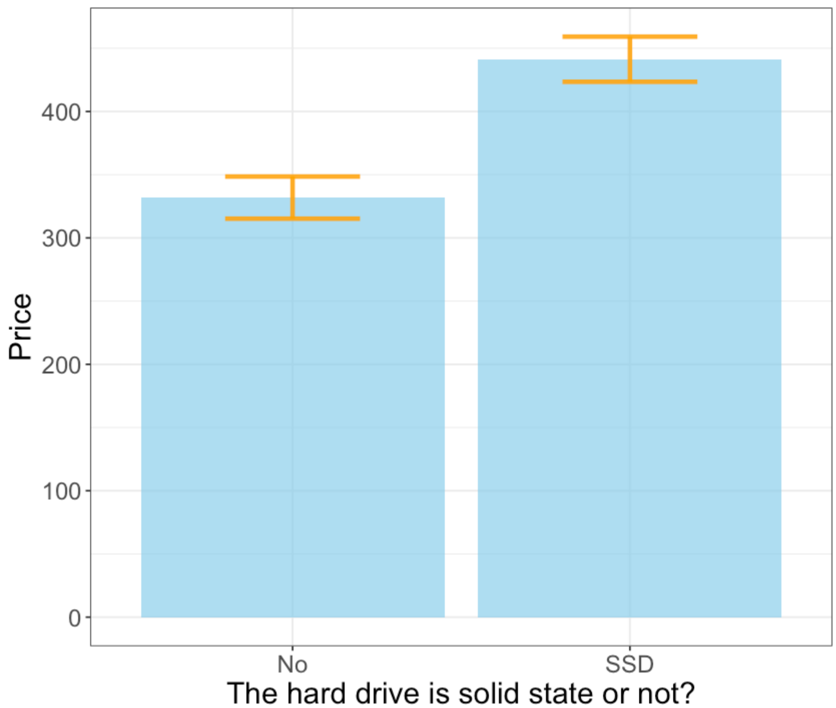
In [41]:
```
t.test(laptop_new$price[which(laptop_new$sale == 'SOLD')], laptop_new$price[
```

```
        Welch Two Sample t-test

data:  laptop_new$price[which(laptop_new$sale == "SOLD")] and laptop_new$pr
ice[which(laptop_new$sale == "NOT SOLD")]
t = -1.6627, df = 89.232, p-value = 0.09988
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -73.032045   6.487173
sample estimates:
mean of x mean of y
 363.6186  396.8910
```

Does price vary with ssd?

In [42]:
```
options(repr.plot.width = 7, repr.plot.height = 6)
price_ssd = laptop_new %>% group_by(ssd) %>%
    summarize(mean_price = mean(price), count = n(), std_price = qt(0.95, n(
price_ssd$std_price[which(price_ssd$count <= 5)] = NA
price_ssd %>% ggplot() +
    geom_bar(aes(x=ssd, y=mean_price),
             stat="identity", fill="skyblue", alpha=0.7) +
    geom_errorbar(aes(x=ssd, ymin=mean_price-std_price, ymax=mean_price+std_
                  width=0.4, colour="orange", alpha=0.9, size=1.3) +
    theme_bw() + theme(text = element_text(size = 18)) +
    xlab("The hard drive is solid state or not?") + ylab("Price")
```



In [43]:
```
t.test(laptop_new$price[which(laptop_new$ssd == 'SSD')], laptop_new$price[wh
```

```
        Welch Two Sample t-test

data:  laptop_new$price[which(laptop_new$ssd == "SSD")] and laptop_new$pric
e[which(laptop_new$ssd == "No")]
t = 7.4417, df = 191.64, p-value = 3.264e-12
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  80.44973 138.47588
sample estimates:
mean of x mean of y
 441.3269  331.8641
```

Based on the exploratory data analysis, price is higher for laptop that

- has higher spped of precessor;
- has 4GB or 8GB ram;
- lower amount of hard drive space;
- can be bought now (not auction);
- has solid state hard drive.

3. Model Construction

In [44]:
```
colnames(laptop_new)
```

'sale' · 'price' · 'ghz' · 'ram' · 'hd' · 'ssd' · 'BIN' · 'hdc'

3.1 Gamma Regression Model

In [45]:
```
glm.Gamma.log <- glm(formula = price ~ sale + ghz + ram + hd + ssd + BIN,
                         family  = Gamma(link = "log"),
                         data    = laptop_new)
summary(glm.Gamma.log)
```

```
Call:
glm(formula = price ~ sale + ghz + ram + hd + ssd + BIN, family = Gamma(lin
k = "log"),
    data = laptop_new)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.04770  -0.15287   0.02943   0.13668   0.89288

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.3087125  0.4960050   8.687 1.11e-15 ***
saleSOLD    -0.1115059  0.0451550  -2.469  0.01434 *
ghz          0.5266685  0.1964040   2.682  0.00792 **
ram          0.0146952  0.0105692   1.390  0.16590
hd           0.0002124  0.0002371   0.896  0.37142
ssdSSD       0.2827572  0.0505398   5.595 6.91e-08 ***
BINTRUE      0.1700154  0.0405254   4.195 4.03e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for Gamma family taken to be 0.07869769)

    Null deviance: 26.075  on 214  degrees of freedom
Residual deviance: 18.809  on 208  degrees of freedom
AIC: 2625.2

Number of Fisher Scoring iterations: 5
```

In [46]:
```
AIC(glm.Gamma.log)
```

2625.2117672118

In [47]:
```
s = summary(glm.Gamma.log)
coef = data.frame(s$coefficients[-1, ])
colnames(coef) = c('estimate', 'stderr', 't.value', 'p.value')
coef.gamma = coef %>% transmute(est = round(exp(estimate), 2),
                                lwr = round(exp(estimate + qt(0.025, 208)*st
                                upr = round(exp(estimate + qt(0.975, 208)*st
                                p.value = round(p.value, 2))
```

3.2 Log-Normal Regression Model

In [48]:
```
lm.log = lm(log(price) ~ sale +  ghz + ram + hd + ssd + BIN, data = laptop_r
summary(lm.log)
```

```
Call:
lm(formula = log(price) ~ sale + ghz + ram + hd + ssd + BIN,
    data = laptop_new)

Residuals:
     Min       1Q   Median       3Q      Max
-1.19250 -0.13354  0.05807  0.18320  0.82509

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.3847826  0.5593716   7.839 2.32e-13 ***
saleSOLD    -0.1157865  0.0509237  -2.274  0.02400 *
ghz          0.4786274  0.2214954   2.161  0.03185 *
ram          0.0147757  0.0119195   1.240  0.21651
hd           0.0002128  0.0002674   0.796  0.42691
ssdSSD       0.3305850  0.0569965   5.800 2.44e-08 ***
BINTRUE      0.1476478  0.0457027   3.231  0.00144 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3164 on 208 degrees of freedom
Multiple R-squared:  0.2794,	Adjusted R-squared:  0.2586
F-statistic: 13.44 on 6 and 208 DF,  p-value: 7.204e-13
```

In [49]:
```
AIC(lm.log)
```

124.164720345699

In [50]:
```
s = summary(lm.log)
coef = data.frame(s$coefficients[-1, ])
colnames(coef) = c('estimate', 'stderr', 't.value', 'p.value')
coef.log = coef %>% transmute(est = round(exp(estimate), 2),
                              lwr = round(exp(estimate + qt(0.025, 207)*stde
                              upr = round(exp(estimate + qt(0.975, 207)*stde
                              p.value = round(p.value, 2))
```

In [51]:
```
cbind(coef.log, coef.gamma)
```

A data.frame: 6 × 8

	est	lwr	upr	p.value	est	lwr	upr	p.value
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
saleSOLD	0.89	0.81	0.98	0.02	0.89	0.82	0.98	0.01
ghz	1.61	1.04	2.50	0.03	1.69	1.15	2.49	0.01
ram	1.01	0.99	1.04	0.22	1.01	0.99	1.04	0.17
hd	1.00	1.00	1.00	0.43	1.00	1.00	1.00	0.37
ssdSSD	1.39	1.24	1.56	0.00	1.33	1.20	1.47	0.00
BINTRUE	1.16	1.06	1.27	0.00	1.19	1.09	1.28	0.00

The cofidence intervals and P-value given by gamma distribution and log-normal distribution are highly similar.

4. Laptops That Have Not Yet Sold

4.1 Lowest Price Satisfying Requirements

In [52]:
```
subset(laptop_new, hd >= 192 & ssd == "SSD" & sale == "NOT SOLD" & BIN == "T
```

A data.frame: 1 × 8

	sale	price	ghz	ram	hd	ssd	BIN	hdc
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<fct>	<fct>	<fct>
23	NOT SOLD	565	2.5	8	240	SSD	TRUE	192 ~ 300

In [53]:
```
subset(laptop_new, hd >= 192 & ssd == "SSD" & sale == "NOT SOLD")
```

A data.frame: 1 × 8

	sale	price	ghz	ram	hd	ssd	BIN	hdc
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<fct>	<fct>	<fct>
23	NOT SOLD	565	2.5	8	240	SSD	TRUE	192 ~ 300

In [54]:
```
laptop[23, ]
```

A data.frame: 1 × 7

	sale	price	ghz	ram	hd	ssd	BIN
	<fct>	<dbl>	<dbl>	<int>	<int>	<fct>	<fct>
23	NOT SOLD	565	2.5	8	240	SSD	TRUE

In [55]:
```
quantile(laptop_new$hd)
```

0%: 80 **25%:** 128 **50%:** 192 **75%:** 303.5 **100%:** 512

In [56]:
```
subset(laptop_new, hd >= 128 & ssd == "SSD" & sale == "NOT SOLD") %>% arrang
```

A data.frame: 13 × 8

sale	price	ghz	ram	hd	ssd	BIN	hdc
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<fct>	<fct>	<fct>
NOT SOLD	373.00	2.5	8	128	SSD	FALSE	80 ~ 128
NOT SOLD	373.00	2.5	8	128	SSD	FALSE	80 ~ 128
NOT SOLD	410.00	2.6	8	128	SSD	TRUE	80 ~ 128
NOT SOLD	437.71	2.5	4	128	SSD	TRUE	80 ~ 128
NOT SOLD	437.71	2.5	4	128	SSD	TRUE	80 ~ 128
NOT SOLD	499.99	2.7	8	160	SSD	FALSE	128 ~ 192
NOT SOLD	500.00	2.7	8	160	SSD	FALSE	128 ~ 192
NOT SOLD	560.00	2.7	8	160	SSD	FALSE	128 ~ 192
NOT SOLD	564.95	2.5	8	128	SSD	TRUE	80 ~ 128
NOT SOLD	565.00	2.5	8	240	SSD	TRUE	192 ~ 300
NOT SOLD	565.00	2.7	8	160	SSD	FALSE	128 ~ 192
NOT SOLD	579.00	2.7	8	160	SSD	FALSE	128 ~ 192
NOT SOLD	699.95	2.5	4	128	SSD	TRUE	80 ~ 128

In [57]:
```
idx = rownames(subset(laptop_new, hd >= 128 & ssd == "SSD" & sale == "NOT SC
laptop[idx, ] %>% arrange(price)
```

A data.frame: 13 × 7

sale	price	ghz	ram	hd	ssd	BIN
<fct>	<dbl>	<dbl>	<int>	<int>	<fct>	<fct>
NOT SOLD	373.00	2.5	8	128	SSD	FALSE
NOT SOLD	373.00	2.5	8	128	SSD	FALSE
NOT SOLD	410.00	2.6	8	128	SSD	TRUE
NOT SOLD	437.71	2.5	4	128	SSD	TRUE
NOT SOLD	437.71	2.5	4	128	SSD	TRUE
NOT SOLD	499.99	2.7	8	NA	SSD	FALSE
NOT SOLD	500.00	2.7	8	160	SSD	FALSE
NOT SOLD	560.00	2.7	8	160	SSD	FALSE
NOT SOLD	564.95	2.5	8	128	SSD	TRUE
NOT SOLD	565.00	2.5	8	240	SSD	TRUE
NOT SOLD	565.00	2.7	8	160	SSD	FALSE
NOT SOLD	579.00	2.7	8	160	SSD	FALSE
NOT SOLD	699.95	2.5	4	128	SSD	TRUE

4.2 Cost-Optimal Laptop Based On Model

In [58]:
```
sold = subset(laptop_new, !(ram %in% c(12, 16)) & laptop_new$sale=="SOLD")
not_sold = subset(laptop_new, !(ram %in% c(12, 16)) & laptop_new$sale=="NOT
```

In [59]:
```
# log-normal
log.normal = lm(log(price) ~ ghz + ram + hd + ssd + BIN, data = sold)
mse = sum(exp(predict(log.normal, sold)) - sold$price)^2/nrow(sold)
mse
```

30029.2337176788

In [60]:
```
# gamma regression
gamma.reg = glm(formula = price ~ ghz + ram + hd + ssd + BIN, family  = Gamm
mse = sum(exp(predict(gamma.reg, sold)) - sold$price)^2/nrow(sold)
mse
```

0.0857564501835868

In [61]:
```
# random forest
rf = randomForest(price ~ ghz + ram + hd + ssd + BIN, data = sold,
                  mtry = 3, ntree = 500, importance = TRUE, na.action = na.c
```

In [62]:
```
print(rf)
```

```
Call:
 randomForest(formula = price ~ ghz + ram + hd + ssd + BIN, data = sold,
mtry = 3, ntree = 500, importance = TRUE, na.action = na.omit)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 3

          Mean of squared residuals: 7858.882
                    % Var explained: 41.13
```

In [63]:
```
not_sold$pred_price = exp(predict(gamma.reg, not_sold))
not_sold %>% mutate(diff = price - pred_price) %>%
    filter(hd >= 128 & ssd == "SSD" & sale == "NOT SOLD") %>%
    arrange(diff)
```

A data.frame: 13 × 10

sale	price	ghz	ram	hd	ssd	BIN	hdc	pred_price	diff
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<fct>	<fct>	<fct>	<dbl>	<dbl>
NOT SOLD	410.00	2.6	8	128	SSD	TRUE	80 ~ 128	476.6248	-66.624827
NOT SOLD	373.00	2.5	8	128	SSD	FALSE	80 ~ 128	388.9711	-15.971126
NOT SOLD	373.00	2.5	8	128	SSD	FALSE	80 ~ 128	388.9711	-15.971126
NOT SOLD	437.71	2.5	4	128	SSD	TRUE	80 ~ 128	428.5822	9.127776
NOT SOLD	437.71	2.5	4	128	SSD	TRUE	80 ~ 128	428.5822	9.127776
NOT SOLD	499.99	2.7	8	160	SSD	FALSE	128 ~ 192	417.3469	82.643127
NOT SOLD	500.00	2.7	8	160	SSD	FALSE	128 ~ 192	417.3469	82.653127
NOT SOLD	564.95	2.5	8	128	SSD	TRUE	80 ~ 128	459.7282	105.221804
NOT SOLD	565.00	2.5	8	240	SSD	TRUE	192 ~ 300	456.8796	108.120371
NOT SOLD	560.00	2.7	8	160	SSD	FALSE	128 ~ 192	417.3469	142.653127
NOT SOLD	565.00	2.7	8	160	SSD	FALSE	128 ~ 192	417.3469	147.653127
NOT SOLD	579.00	2.7	8	160	SSD	FALSE	128 ~ 192	417.3469	161.653127
NOT SOLD	699.95	2.5	4	128	SSD	TRUE	80 ~ 128	428.5822	271.367776

In [ ]:

In [64]:
```
options(warn = 0)
```

In [ ]:

In [ ]: