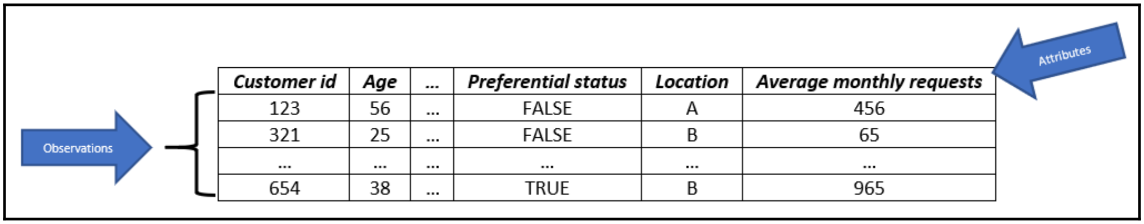
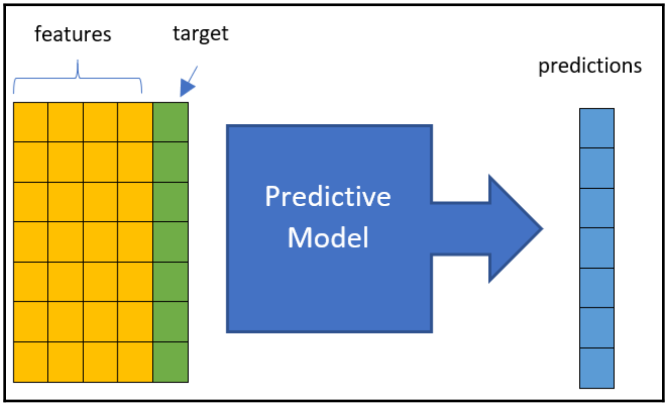
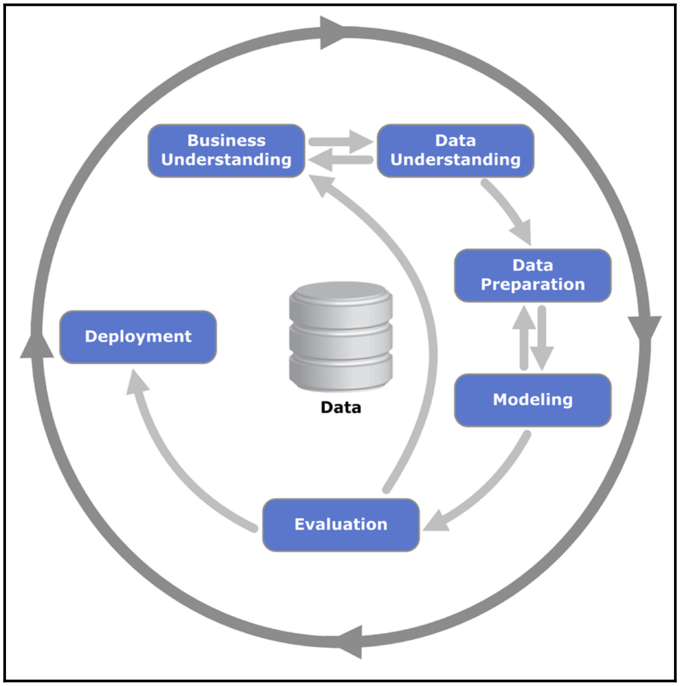
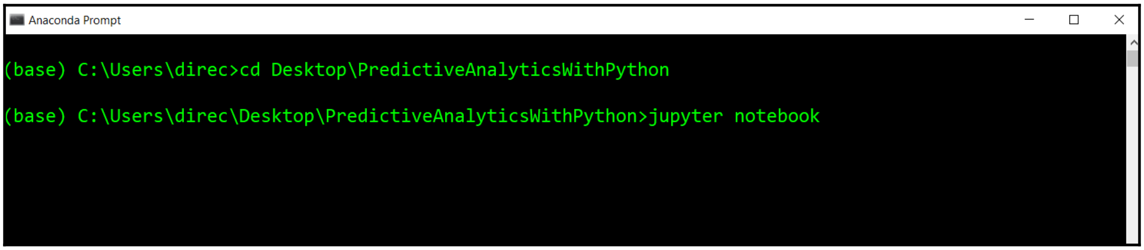
Chapter 01: The Predictive Analytics Process



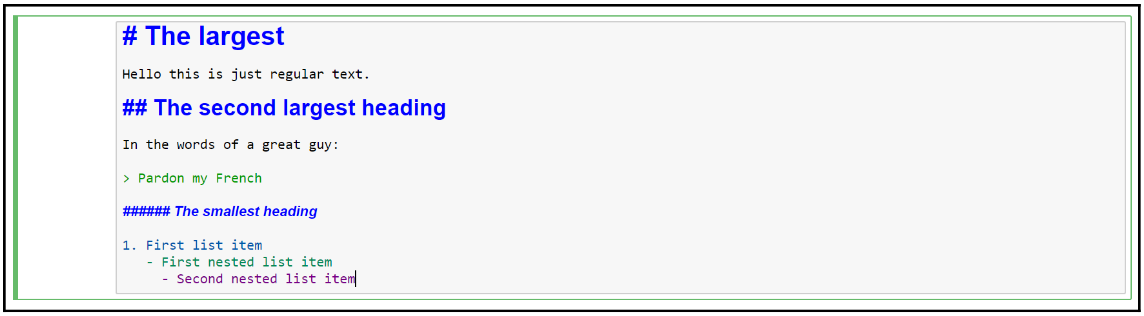
Customer id	*Age*	*...*	*Preferential status*	*Location*	*Average monthly requests*
123	56	...	FALSE	A	456
321	25	...	FALSE	B	65
...	...	...	...	...	...
654	38	...	TRUE	B	965




Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment
Data


Anaconda Prompt
(base) C:\Users\direc>cd Desktop\PredictiveAnalyticsWithPython
(base) C:\Users\direc\Desktop\PredictiveAnalyticsWithPython>jupyter notebook

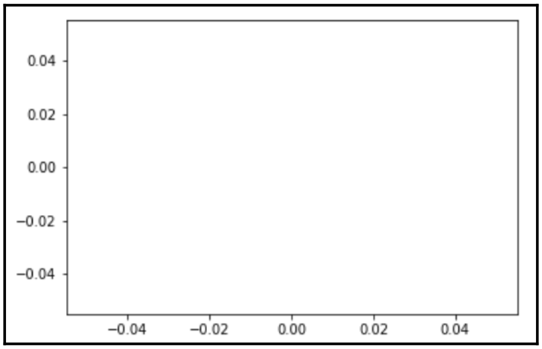

Home
localhost:8888/tree
jupyter
Quit
Logout
Files
Running
Clusters
Select items to perform actions on them.
Upload
New
Name
The notebook list is empty.
Notebook:
Python 3
Other:
Text File
Folder
Terminal

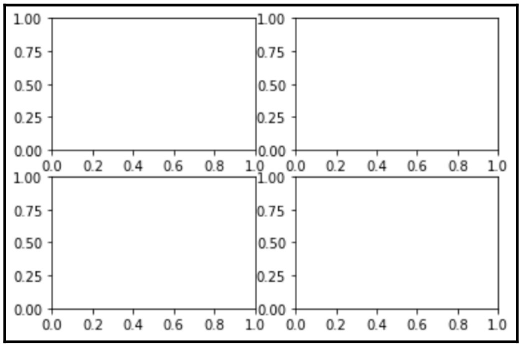

The largest
Hello this is just regular text.
The second largest heading
In the words of a great guy:
> Pardon my French
The smallest heading
1. First list item
- First nested list item
- Second nested list item

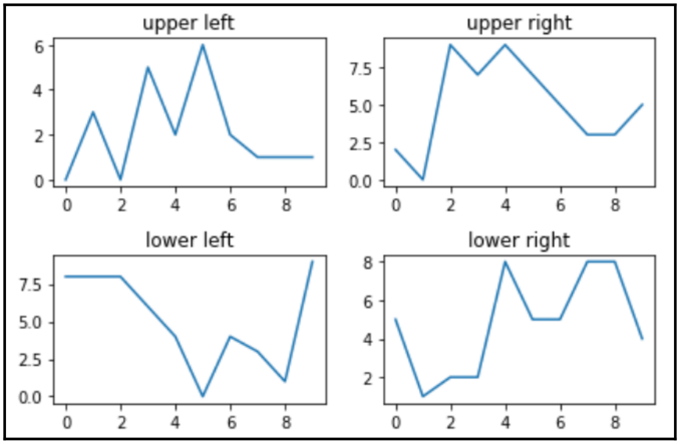

The largest
Hello this is just regular text.
The second largest heading
In the words of a great guy:
Pardon my French
The smallest heading
1. First list item
First nested list item
Second nested list item

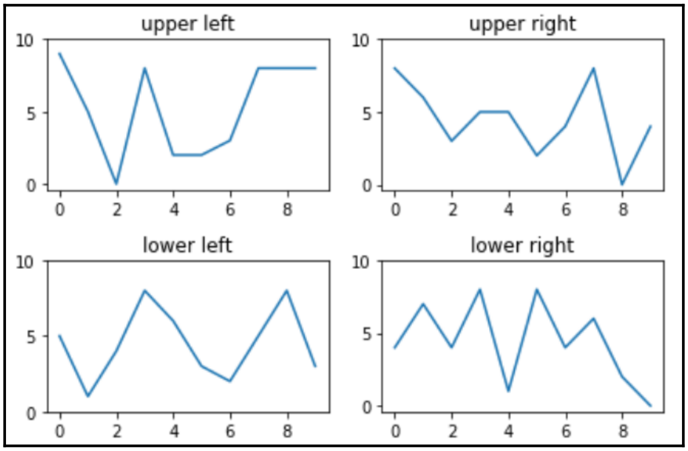
```
In [2]: x = 7
        x ** 2

Out[2]: 49
```

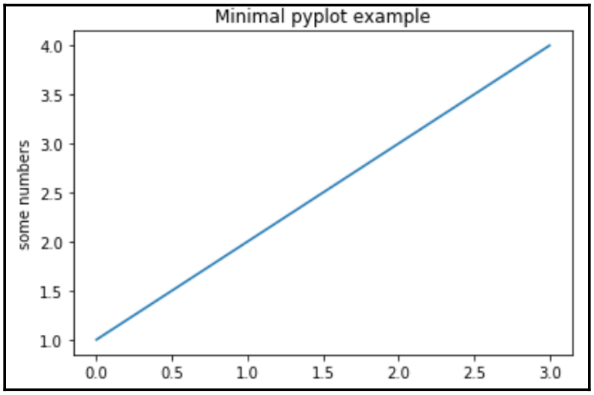
```
In [1]: for i in range(10):
            print(str(i) + ' squared is ' + str(i**2))

        0 squared is 0
        1 squared is 1
        2 squared is 4
        3 squared is 9
        4 squared is 16
        5 squared is 25
        6 squared is 36
        7 squared is 49
        8 squared is 64
        9 squared is 81
```






upper left
upper right
lower left
lower right


upper left
upper right
lower left
lower right


Minimal pyplot example
some numbers

Chapter 02: Problem Understanding and Data Preparation

	carat	cut	color	clarity	depth	table	price	x	y	z
0	0.23	Ideal	E	SI2	61.5	55.0	326	3.95	3.98	2.43
1	0.21	Premium	E	SI1	59.8	61.0	326	3.89	3.84	2.31
2	0.23	Good	E	VS1	56.9	65.0	327	4.05	4.07	2.31
3	0.29	Premium	I	VS2	62.4	58.0	334	4.20	4.23	2.63
4	0.31	Good	J	SI2	63.3	58.0	335	4.34	4.35	2.75

	carat	depth	table	price	x	y	z
count	53940.000000	53940.000000	53940.000000	53940.000000	53940.000000	53940.000000	53940.000000
mean	0.797940	61.749405	57.457184	3932.799722	5.731157	5.734526	3.538734
std	0.474011	1.432621	2.234491	3989.439738	1.121761	1.142135	0.705699
min	0.200000	43.000000	43.000000	326.000000	0.000000	0.000000	0.000000
25%	0.400000	61.000000	56.000000	950.000000	4.710000	4.720000	2.910000
50%	0.700000	61.800000	57.000000	2401.000000	5.700000	5.710000	3.530000
75%	1.040000	62.500000	59.000000	5324.250000	6.540000	6.540000	4.040000
max	5.010000	79.000000	95.000000	18823.000000	10.740000	58.900000	31.800000

	carat	cut	color	clarity	depth	table	price	x	y	z
11182	1.07	Ideal	F	SI2	61.6	56.0	4954	0.0	6.62	0.0
11963	1.00	Very Good	H	VS2	63.3	53.0	5139	0.0	0.00	0.0
15951	1.14	Fair	G	VS1	57.5	67.0	6381	0.0	0.00	0.0
24520	1.56	Ideal	G	VS2	62.2	54.0	12800	0.0	0.00	0.0
26243	1.20	Premium	D	VVS1	62.1	59.0	15686	0.0	0.00	0.0
27429	2.25	Premium	H	SI2	62.8	59.0	18034	0.0	0.00	0.0
49556	0.71	Good	F	SI2	64.1	60.0	2130	0.0	0.00	0.0
49557	0.71	Good	F	SI2	64.1	60.0	2130	0.0	0.00	0.0

```
carat         1.07
cut          Ideal
color            F
clarity        SI2
depth         61.6
table           56
price         4954
x                0
y             6.62
z                0
Name: 11182, dtype: object
```

	carat	cut	color	clarity	depth	table	price	x	y	z
11182	1.07	Ideal	F	SI2	61.6	56.0	4954	5.7	6.62	0.0

	carat	cut	color	clarity	depth	table	price	x	y	z
24067	2.00	Premium	H	SI2	58.9	57.0	12210	8.09	58.90	8.06
48410	0.51	Very Good	E	VS1	61.8	54.7	1970	5.12	5.15	31.80
49189	0.51	Ideal	E	VS1	61.8	55.0	2075	5.15	31.80	5.12

	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_0	PAY_2	PAY_3	PAY_4	PAY_5	...	BILL_AMT4	BILL_AMT5	BILL_AMT6
ID														
1	20000	2	2	1	24	2	2	-1	-1	-2	...	0	0	0
2	120000	2	2	2	26	-1	2	0	0	0	...	3272	3455	3261
3	90000	2	2	2	34	0	0	0	0	0	...	14331	14948	15549
4	50000	2	2	1	37	0	0	0	0	0	...	28314	28959	29547
5	50000	1	2	1	57	-1	0	-1	0	0	...	20940	19146	19131

	limit_bal	age
count	30000.000000	30000.000000
mean	167484.322667	35.485500
std	129747.661567	9.217904
min	10000.000000	21.000000
25%	50000.000000	28.000000
50%	140000.000000	34.000000
75%	240000.000000	41.000000
max	1000000.000000	79.000000

	bill_amt1	bill_amt2	bill_amt3	bill_amt4	bill_amt5	bill_amt6
count	30000.0	30000.0	30000.0	30000.0	30000.0	30000.0
mean	51223.0	49179.0	47013.0	43263.0	40311.0	38872.0
std	73636.0	71174.0	69349.0	64333.0	60797.0	59554.0
min	-165580.0	-69777.0	-157264.0	-170000.0	-81334.0	-339603.0
25%	3559.0	2985.0	2666.0	2327.0	1763.0	1256.0
50%	22382.0	21200.0	20088.0	19052.0	18104.0	17071.0
75%	67091.0	64006.0	60165.0	54506.0	50190.0	49198.0
max	964511.0	983931.0	1664089.0	891586.0	927171.0	961664.0

	pay_amt1	pay_amt2	pay_amt3	pay_amt4	pay_amt5	pay_amt6
count	30000.0	30000.0	30000.0	30000.0	30000.0	30000.0
mean	5664.0	5921.0	5226.0	4826.0	4799.0	5216.0
std	16563.0	23041.0	17607.0	15666.0	15278.0	17777.0
min	0.0	0.0	0.0	0.0	0.0	0.0
25%	1000.0	833.0	390.0	296.0	252.0	118.0
50%	2100.0	2009.0	1800.0	1500.0	1500.0	1500.0
75%	5006.0	5000.0	4505.0	4013.0	4032.0	4000.0
max	873552.0	1684259.0	896040.0	621000.0	426529.0	528666.0

```
ID
1     0
2     0
3     0
4     0
5     1
6     1
7     1
8     0
9     0
10    1
Name: male, dtype: int32
```

```
0       14
1    10585
2    14030
3     4917
4      123
5      280
6       51
Name: education, dtype: int64
```

```
ID
48      5
70      5
359     4
386     5
449     4
503     6
505     6
1074    6
1266    5
1283    5
1367    4
1370    5
1491    4
1832    6
```

```
1    13713
2    15964
3      323
Name: marriage, dtype: int64
```

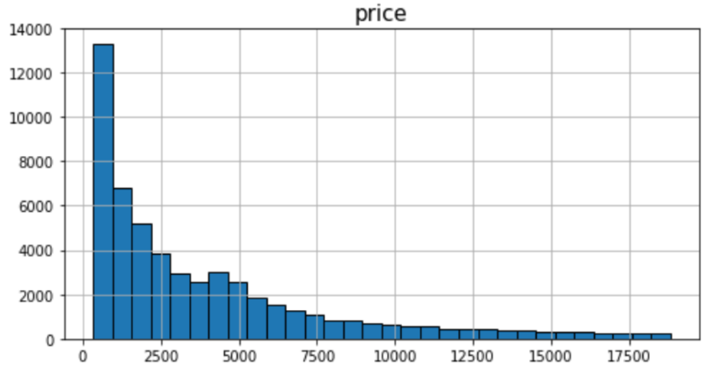
	cut_Fair	cut_Good	cut_Ideal	cut_Premium	cut_Very Good
0	0	0	1	0	0
1	0	0	0	1	0
2	0	1	0	0	0
3	0	0	0	1	0
4	0	1	0	0	0
5	0	0	0	0	1
6	0	0	0	0	1

	cut_Good	cut_Ideal	cut_Premium	cut_Very Good
0	0	1	0	0
1	0	0	1	0
2	1	0	0	0
3	0	0	1	0
4	1	0	0	0
5	0	0	0	1
6	0	0	0	1

```
-2      2759
-1      5686
 0     14737
 1      3688
 2      2667
 3       322
 4        76
 5        26
 6        11
 7         9
 8        19
Name: pay_1, dtype: int64
```

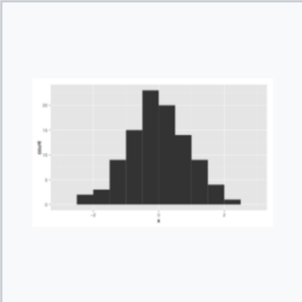
```
delayed_1    0.227267
delayed_2    0.147933
delayed_3    0.140433
delayed_4    0.117000
delayed_5    0.098933
delayed_6    0.102633
dtype: float64
```

Chapter 03: Dataset Understanding – Exploratory Data Analysis

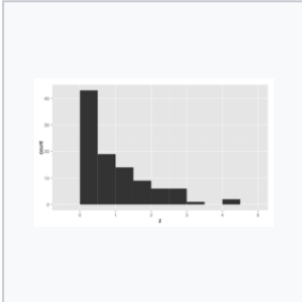



```
count    53930.00
mean      3932.07
std       3988.63
min        326.00
25%        949.25
50%       2401.00
75%       5324.00
max      18823.00
Name: price, dtype: float64
```

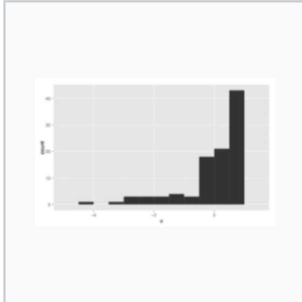
$$\boxed{bin\_size \approx \frac{18823 - 326}{30} \approx 615}$$



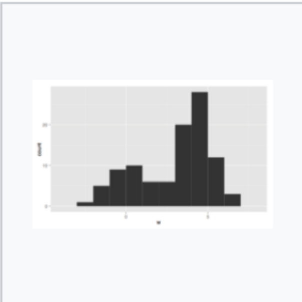
Symmetric, unimodal



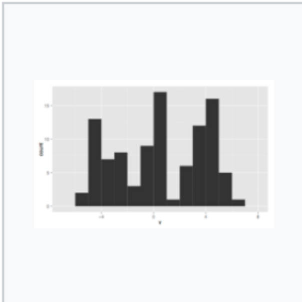
Skewed right



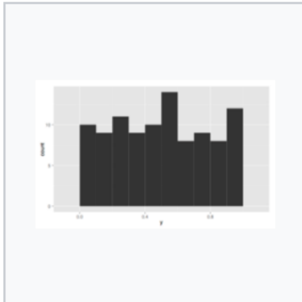
Skewed left



Bimodal

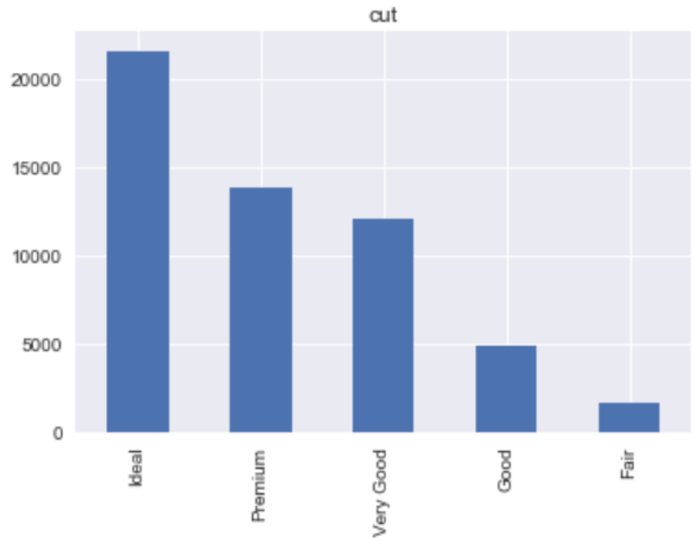


Multimodal

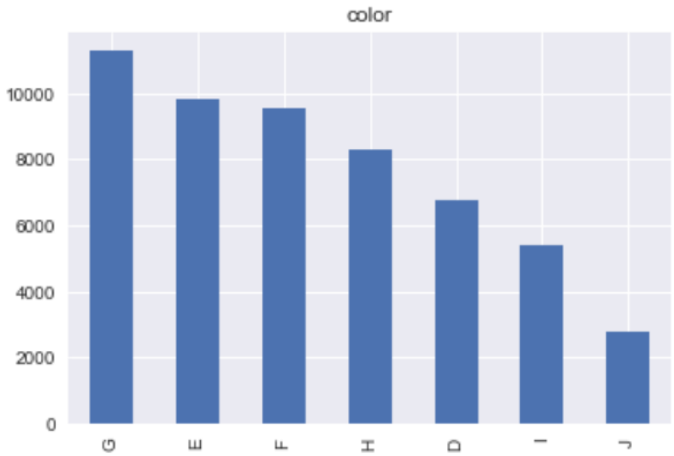


Symmetric

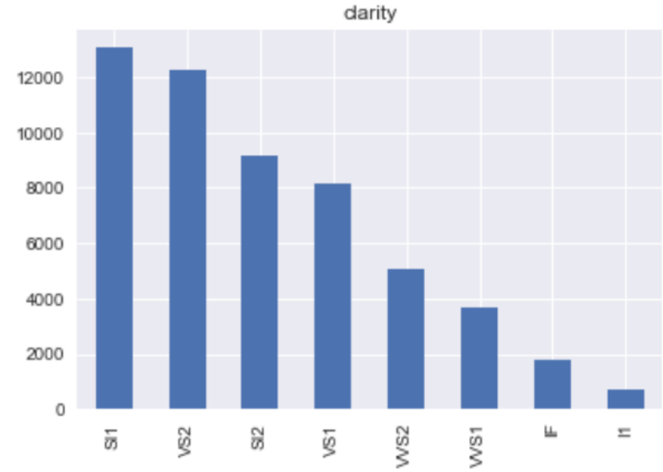
```
           count  percent
Ideal      21549     40.0
Premium    13788     25.6
Very Good  12080     22.4
Good        4904      9.1
Fair        1609      3.0
```

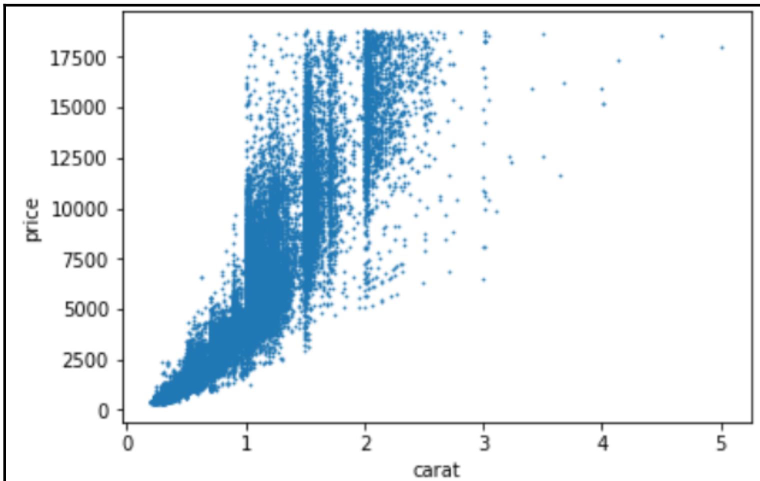


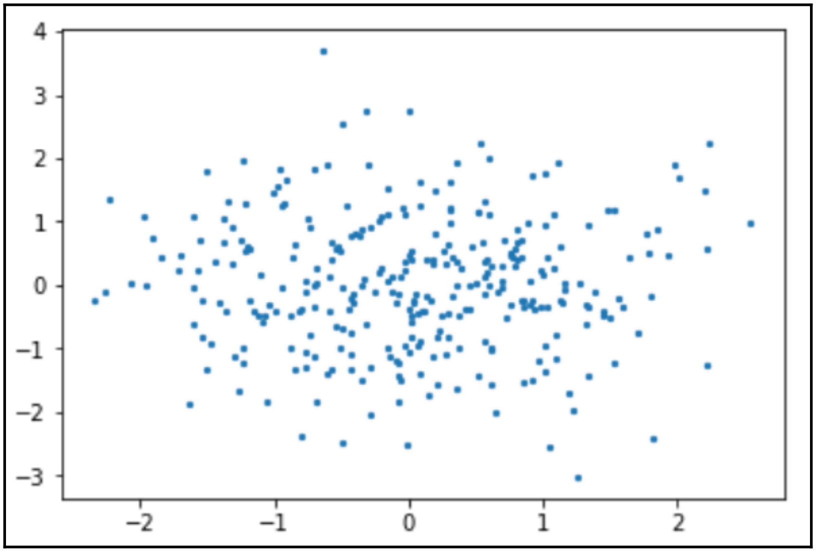
	count	percent
G	11290	20.9
E	9795	18.2
F	9540	17.7
H	8301	15.4
D	6774	12.6
I	5422	10.1
J	2808	5.2



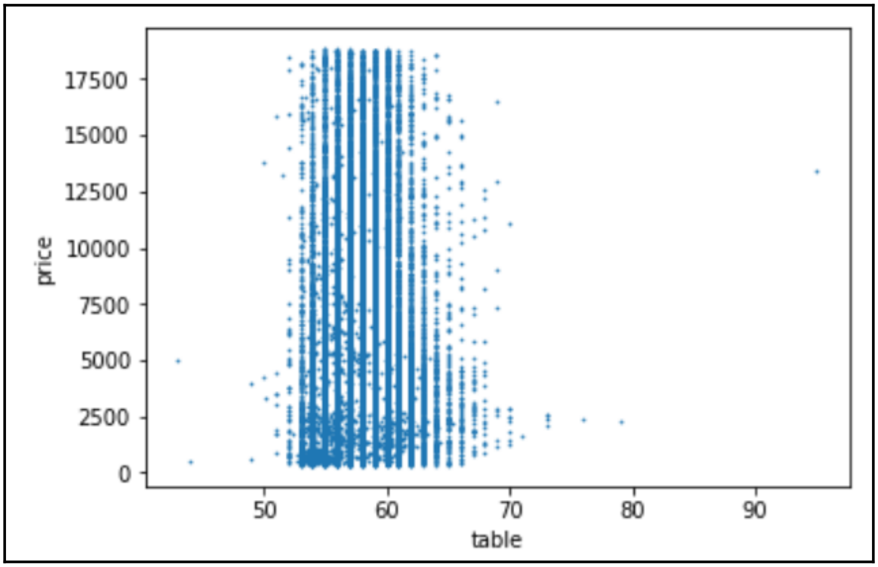
	count	percent
SI1	13065	24.2
VS2	12256	22.7
SI2	9190	17.0
VS1	8168	15.1
VVS2	5066	9.4
VVS1	3654	6.8
IF	1790	3.3
I1	741	1.4



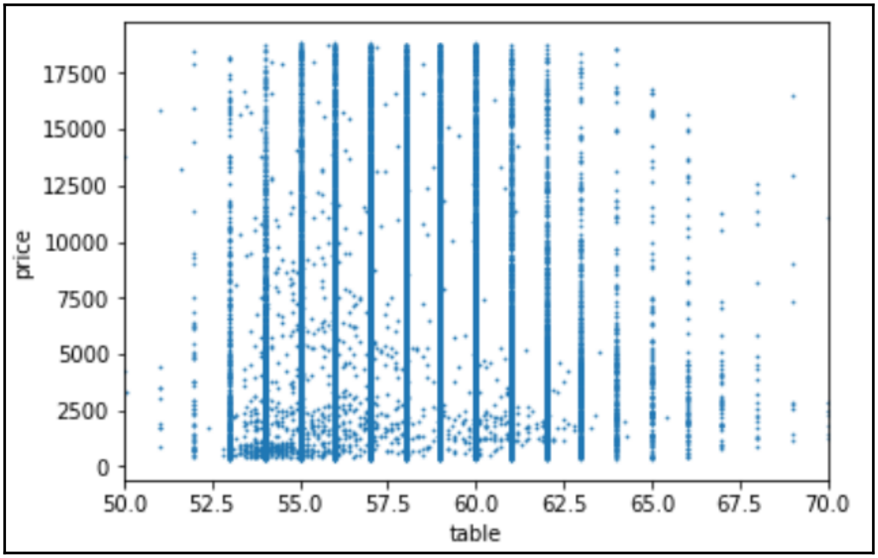


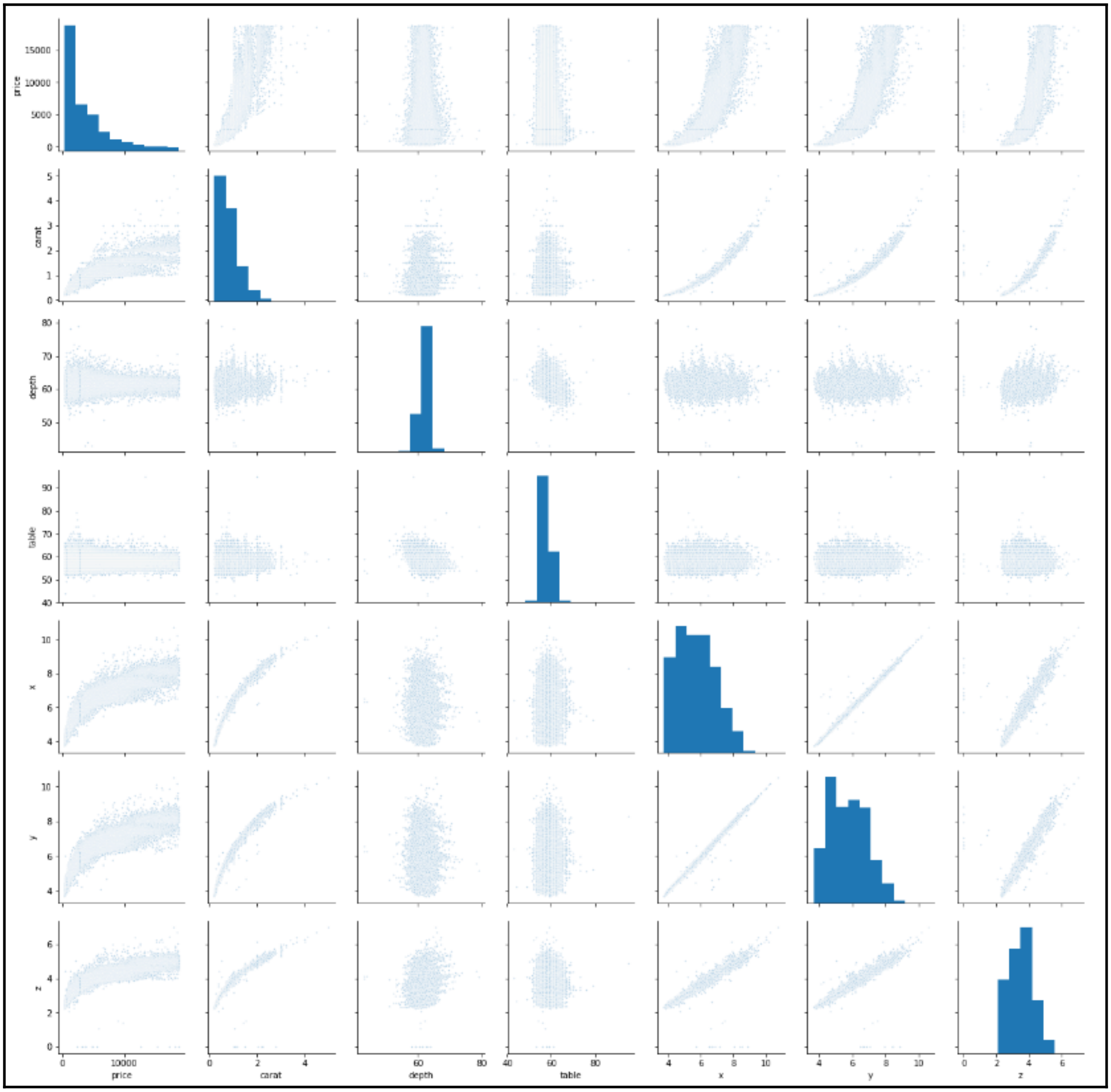


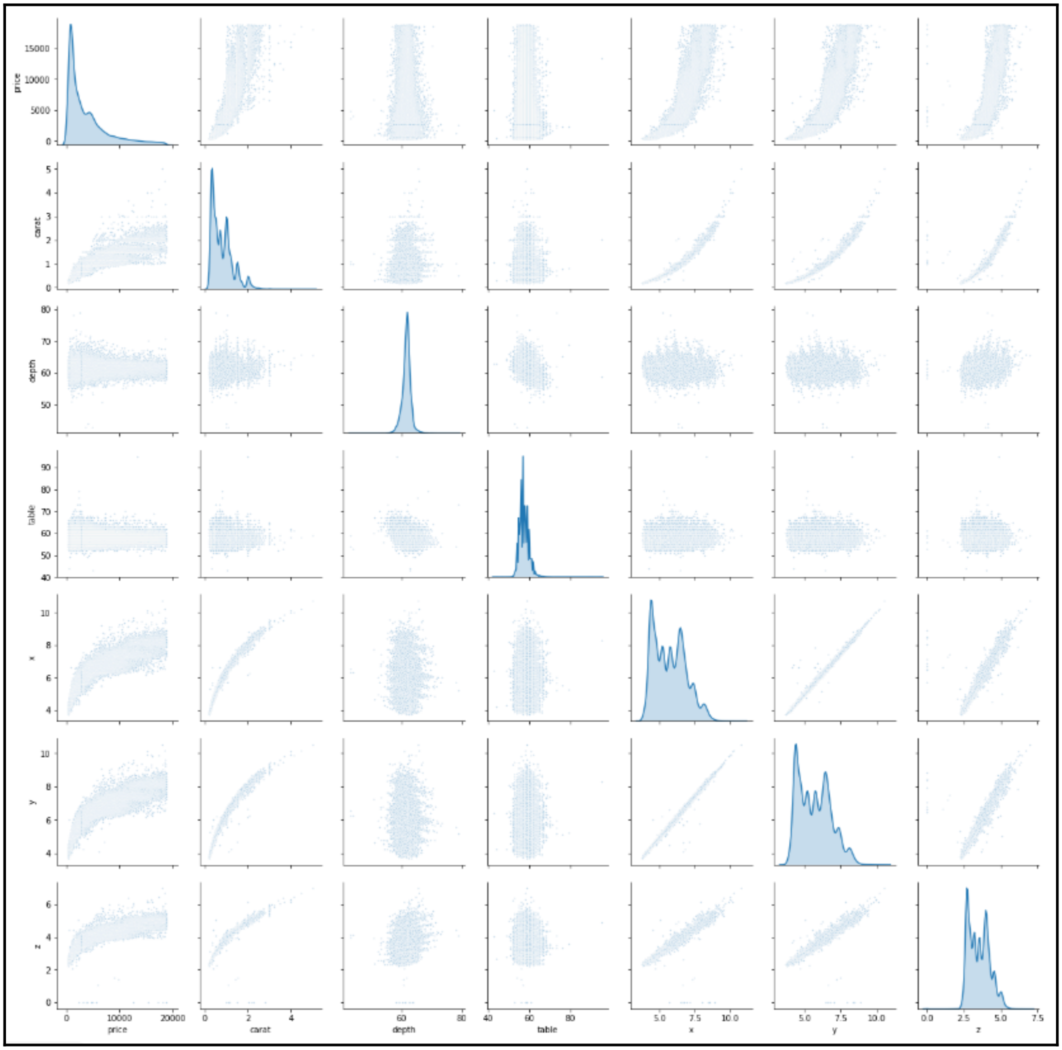
4
3
2
1
0
−1
−2
−3
−2
−1
0
1
2

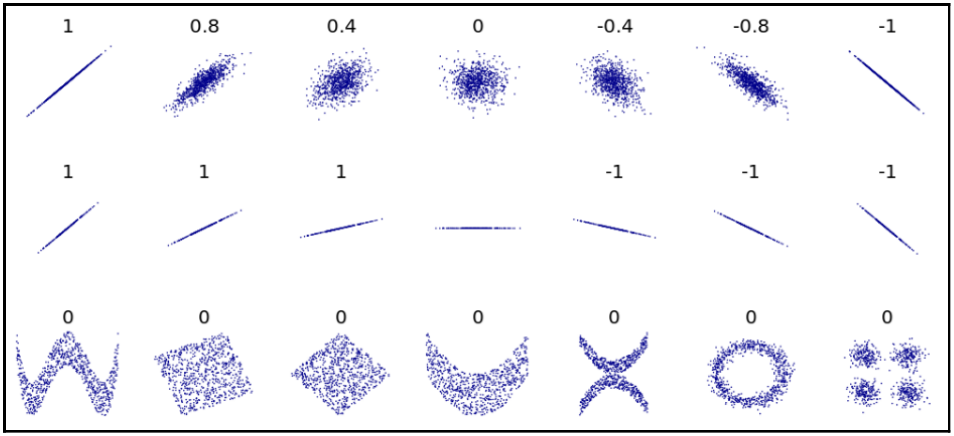


17500
15000
12500
10000
7500
5000
2500
0
price
50
60
70
80
90
table

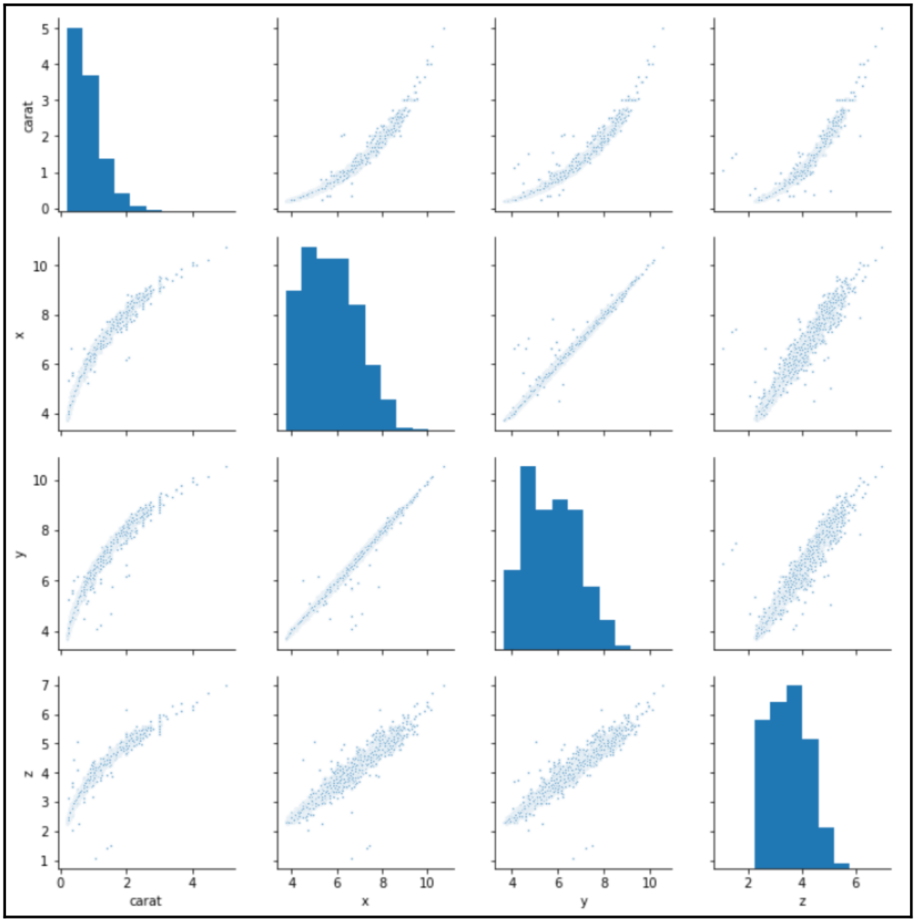

price
17500
15000
12500
10000
7500
5000
2500
0
50.0
52.5
55.0
57.5
60.0
62.5
65.0
67.5
70.0
table


price
carat
depth
table
x
y
z


price
carat
depth
table
x
y
z



	price	carat	depth	table	x	y	z
price	1.000000	0.921603	-0.010595	0.127157	0.887216	0.888810	0.877430
carat	0.921603	1.000000	0.028317	0.181650	0.977761	0.976844	0.970905
depth	-0.010595	0.028317	1.000000	-0.295722	-0.025020	-0.028151	0.097057
table	0.127157	0.181650	-0.295722	1.000000	0.196129	0.189964	0.155012
x	0.887216	0.977761	-0.025020	0.196129	1.000000	0.998652	0.985904
y	0.888810	0.976844	-0.028151	0.189964	0.998652	1.000000	0.985538
z	0.877430	0.970905	0.097057	0.155012	0.985904	0.985538	1.000000

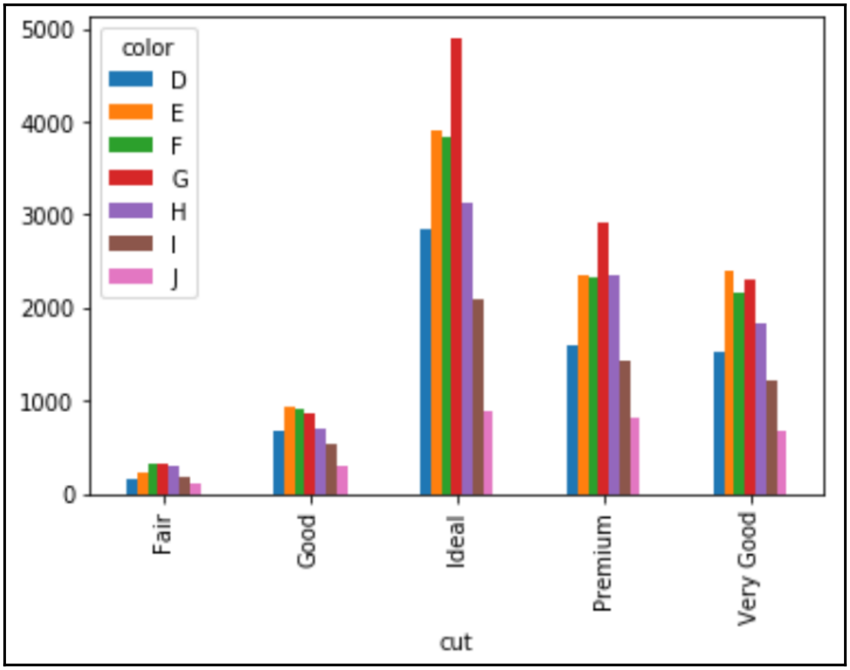


	carat	x	y	z
carat	1.000000	0.977778	0.976860	0.976478
x	0.977778	1.000000	0.998657	0.991077
y	0.976860	0.998657	1.000000	0.990730
z	0.976478	0.991077	0.990730	1.000000

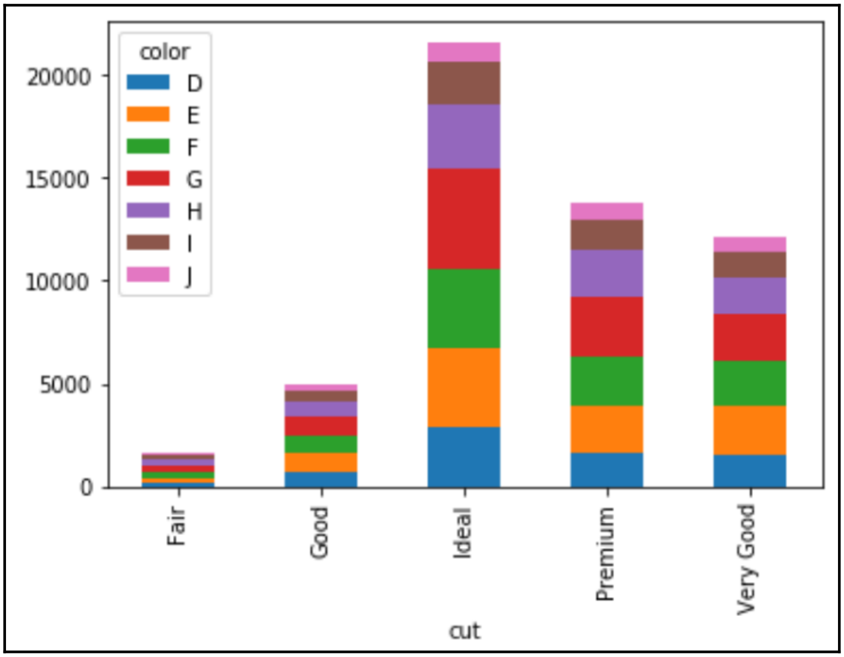
color	D	E	F	G	H	I	J
cut							
Fair	163	224	312	313	303	175	119
Good	662	933	907	871	702	522	307
Ideal	2834	3902	3826	4883	3115	2093	896
Premium	1602	2337	2331	2924	2358	1428	808
Very Good	1513	2399	2164	2299	1823	1204	678

color	D	E	F	G	H	I	J	Total
cut								
Fair	163	224	312	313	303	175	119	1609
Good	662	933	907	871	702	522	307	4904
Ideal	2834	3902	3826	4883	3115	2093	896	21549
Premium	1602	2337	2331	2924	2358	1428	808	13788
Very Good	1513	2399	2164	2299	1823	1204	678	12080
Total	6774	9795	9540	11290	8301	5422	2808	53930

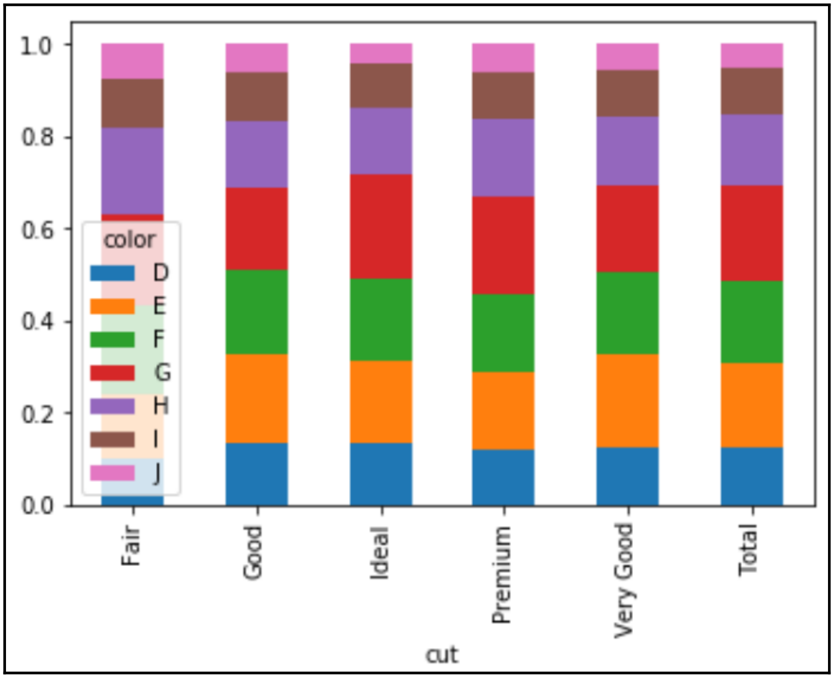
color	D	E	F	G	H	I	J	Total
cut								
Fair	10.1	13.9	19.4	19.5	18.8	10.9	7.4	100.0
Good	13.5	19.0	18.5	17.8	14.3	10.6	6.3	100.0
Ideal	13.2	18.1	17.8	22.7	14.5	9.7	4.2	100.0
Premium	11.6	16.9	16.9	21.2	17.1	10.4	5.9	100.0
Very Good	12.5	19.9	17.9	19.0	15.1	10.0	5.6	100.0
Total	12.6	18.2	17.7	20.9	15.4	10.1	5.2	100.0

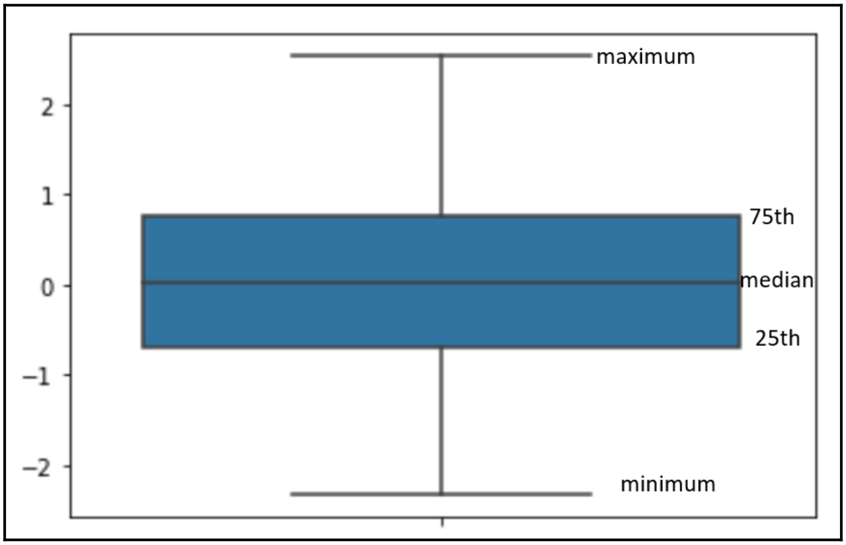


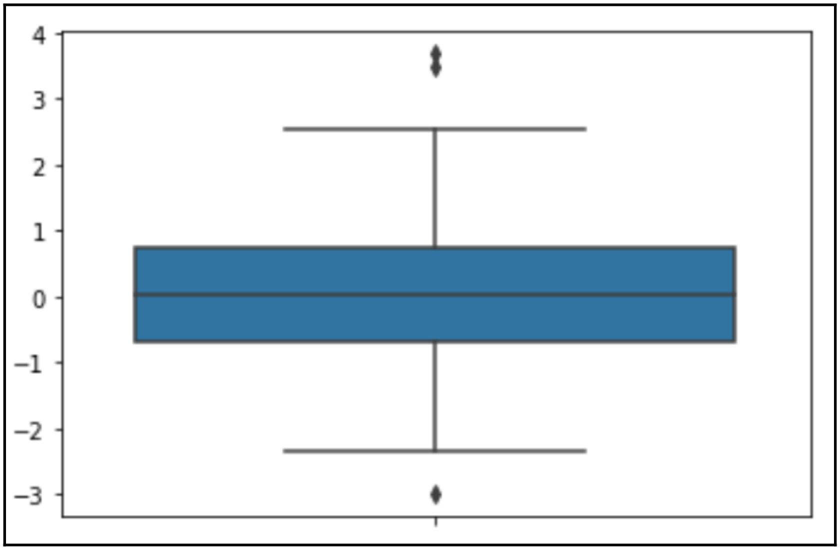
5000
4000
3000
2000
1000
0
color
D
E
F
G
H
I
J
Fair
Good
Ideal
Premium
Very Good
cut

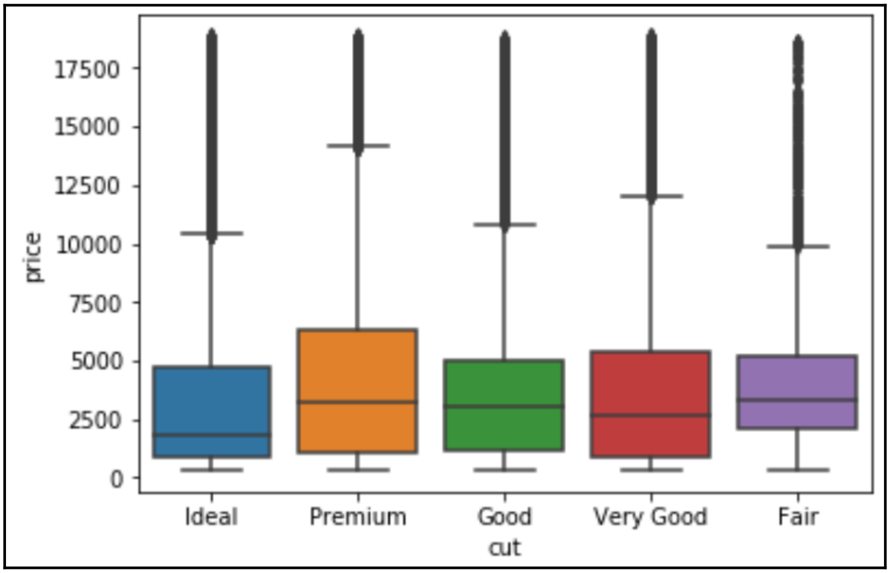


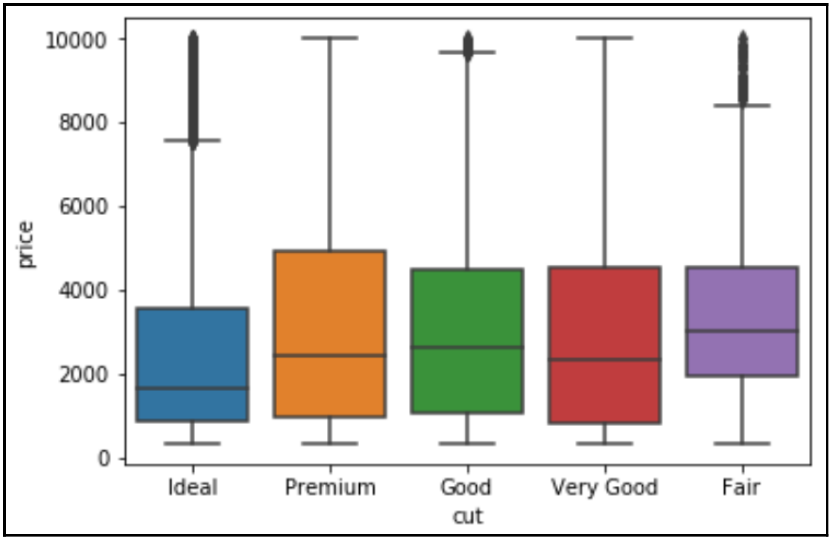
20000
15000
10000
5000
0
color
D
E
F
G
H
I
J
Fair
Good
Ideal
Premium
Very Good
cut


1.0
0.8
0.6
0.4
0.2
0.0
color
D
E
F
G
H
I
J
Fair
Good
Ideal
Premium
Very Good
Total
cut


maximum
2
1
75th
0
median
25th
−1
−2
minimum

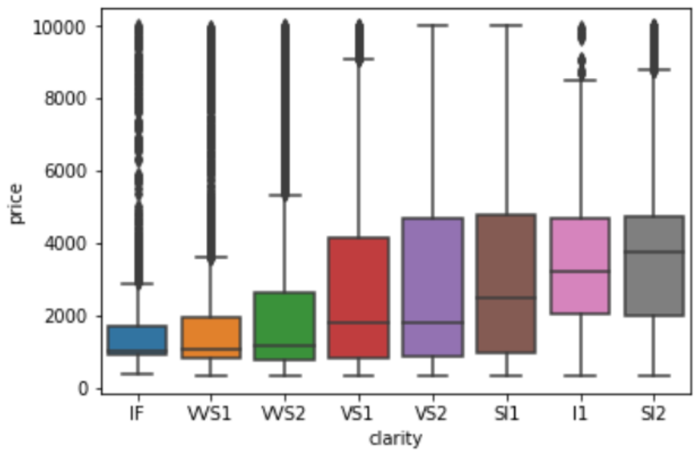

4
3
2
1
0
−1
−2
−3

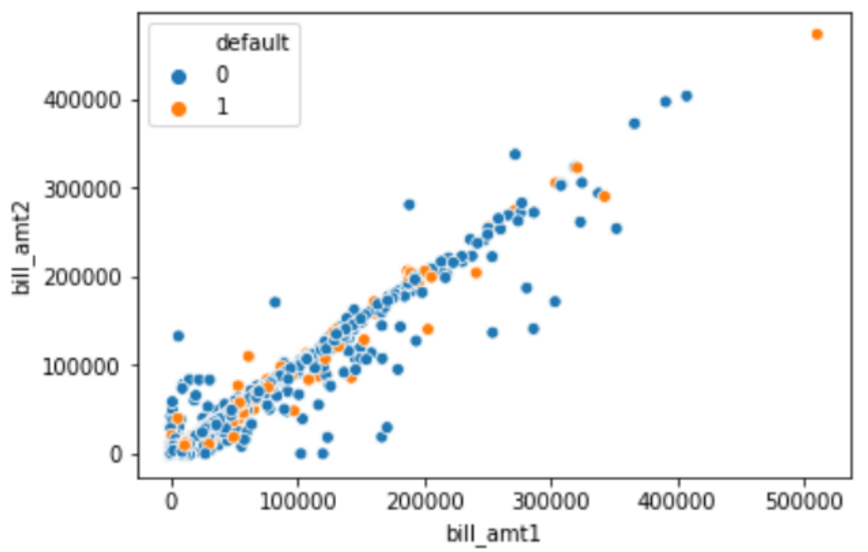

17500
15000
12500
10000
7500
5000
2500
0
price
Ideal
Premium
Good
Very Good
Fair
cut

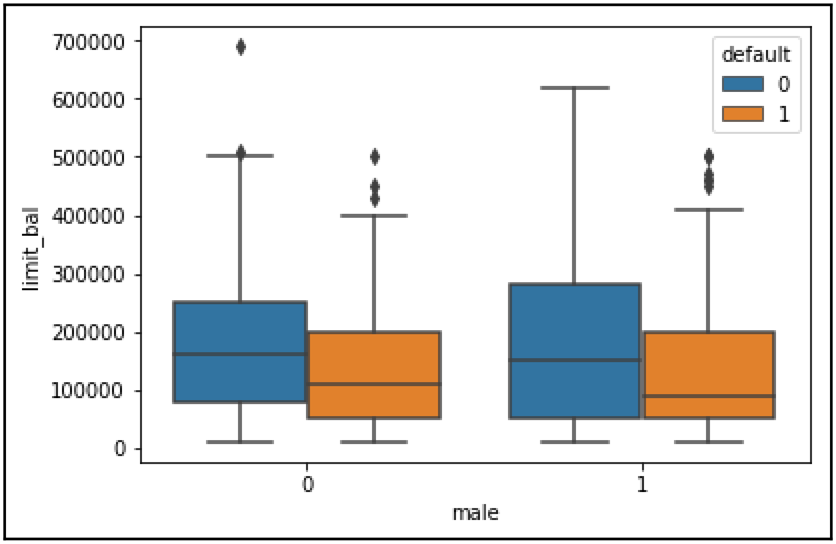


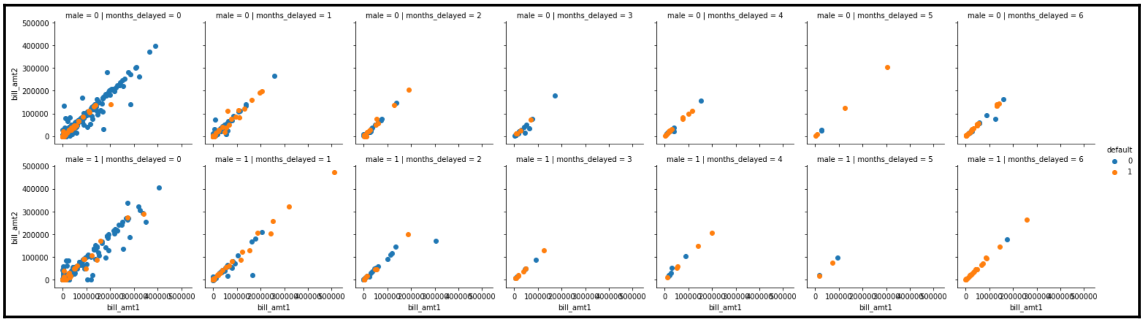

```
cut
Ideal         1810
Very Good     2648
Good          3054
Premium       3183
Fair          3282
Name: price, dtype: int64
```

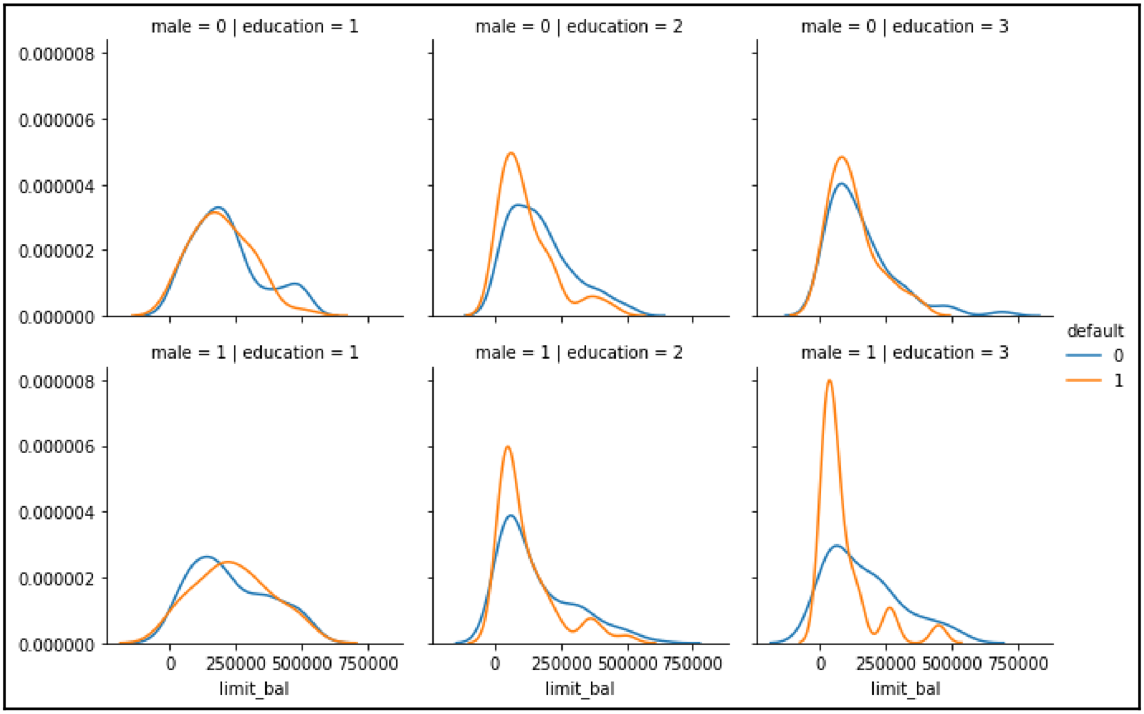
```
clarity
IF       1080.0
VVS1     1092.5
VVS2     1311.0
VS1      2005.0
VS2      2053.0
SI1      2822.0
I1       3344.0
SI2      4072.0
Name: price, dtype: float64
```






default
0
1
limit_bal
700000
600000
500000
400000
300000
200000
100000
0
0
1
male

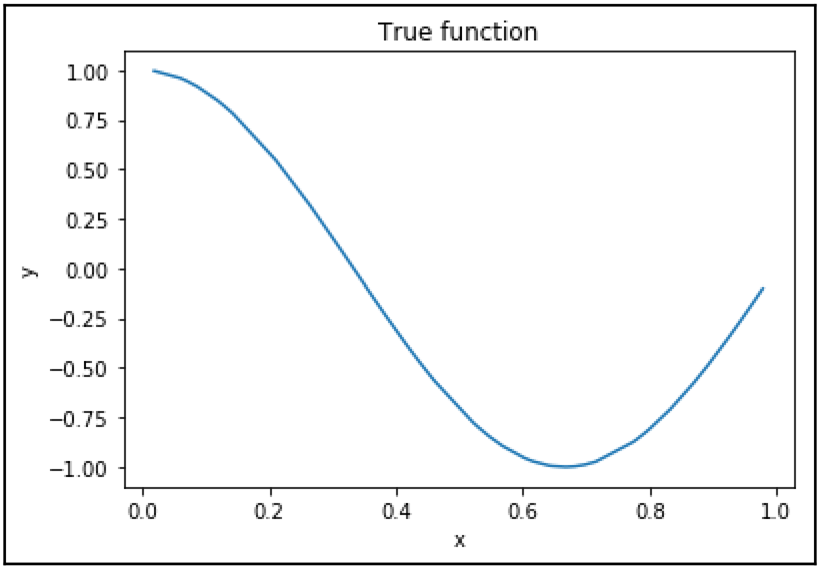

male = 0 | months_delayed = 0
male = 0 | months_delayed = 1
male = 0 | months_delayed = 2
male = 0 | months_delayed = 3
male = 0 | months_delayed = 4
male = 0 | months_delayed = 5
male = 0 | months_delayed = 6
male = 1 | months_delayed = 0
male = 1 | months_delayed = 1
male = 1 | months_delayed = 2
male = 1 | months_delayed = 3
male = 1 | months_delayed = 4
male = 1 | months_delayed = 5
male = 1 | months_delayed = 6
bill_amt2
bill_amt1
default
0
1

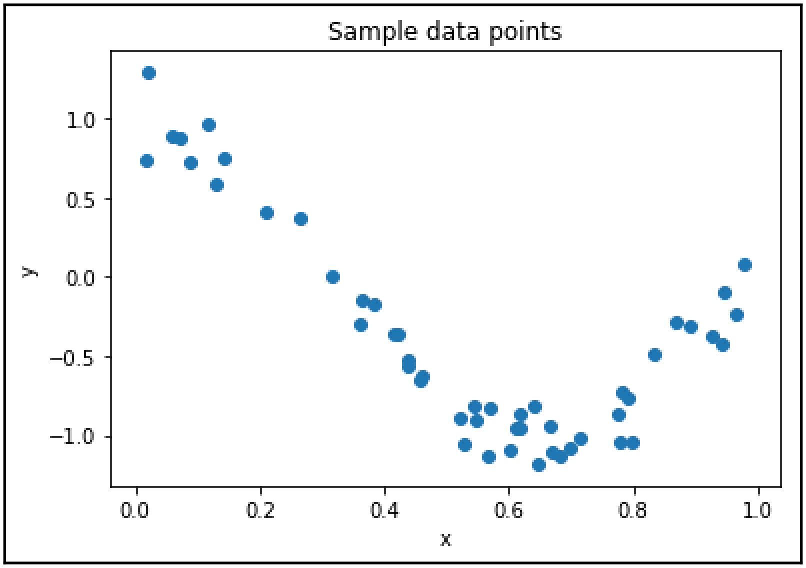


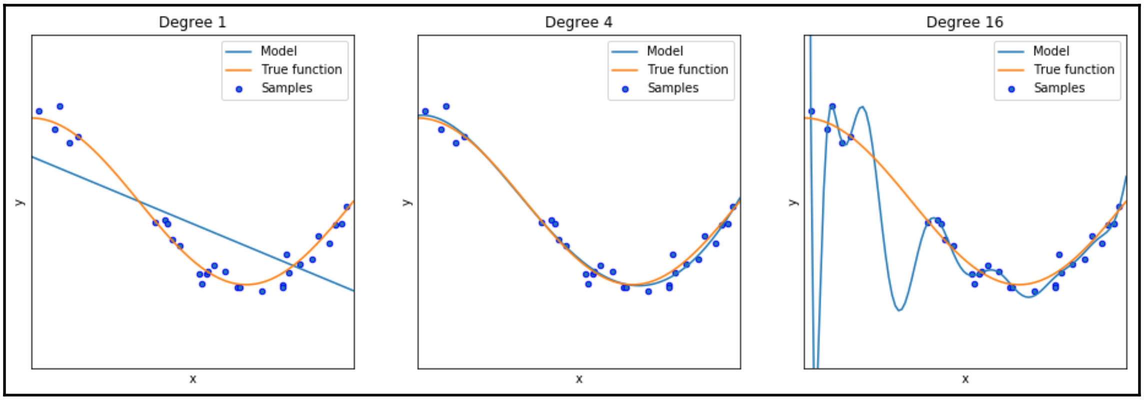
male = 0 | education = 1
male = 0 | education = 2
male = 0 | education = 3
male = 1 | education = 1
male = 1 | education = 2
male = 1 | education = 3
0.000008
0.000006
0.000004
0.000002
0.000000
0
250000
500000
750000
limit_bal
default
0
1

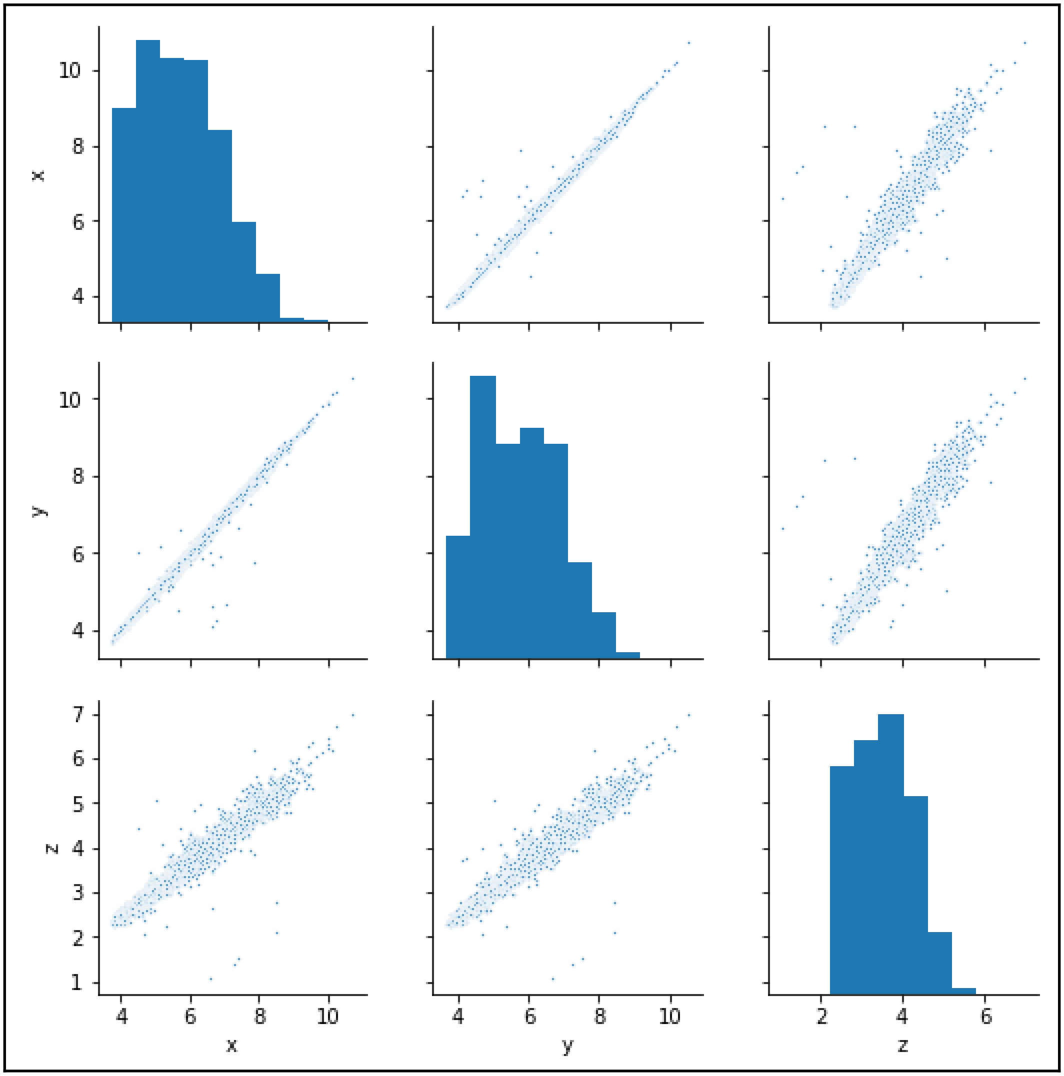
Chapter 04: Predicting Numerical Values with Machine Learning

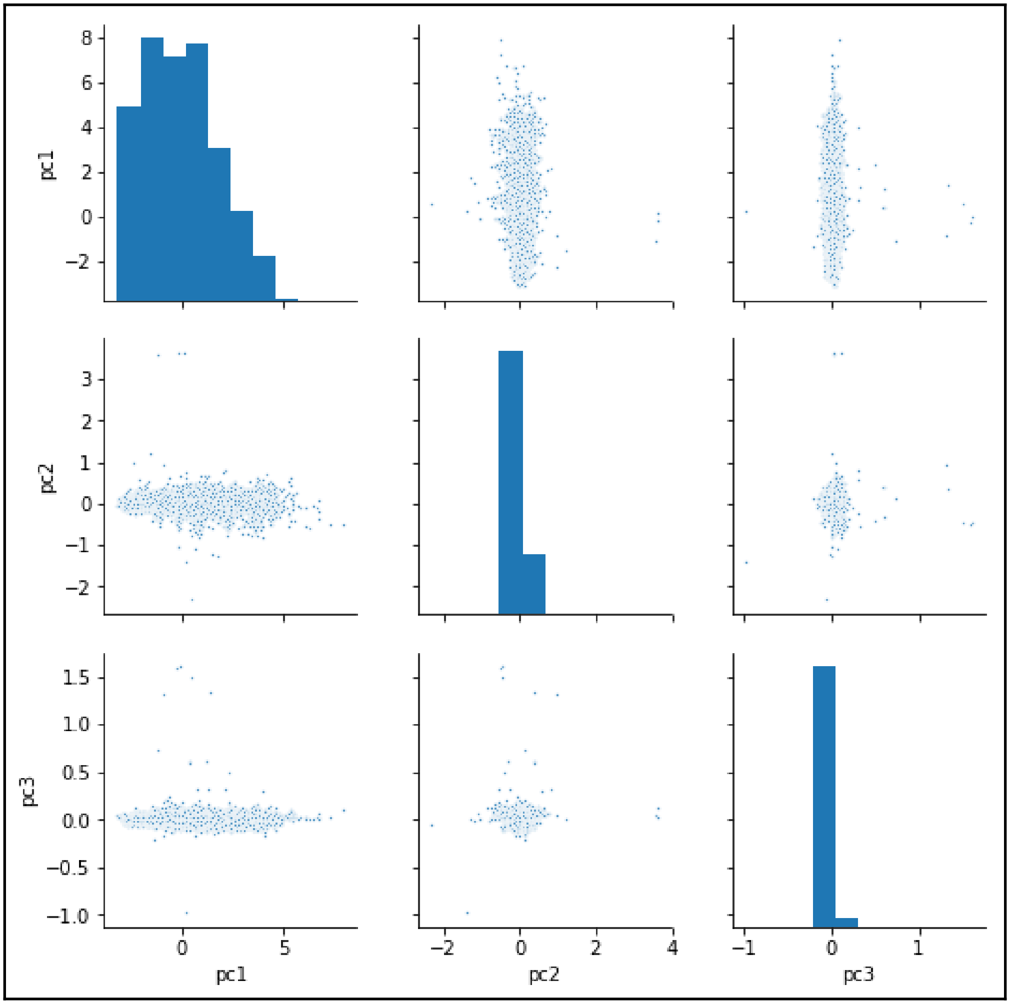
	Carat	Predicted price
0	0.5	2004.012
1	1.0	4008.024
2	1.5	6012.036
3	2.0	8016.048
4	2.5	10020.060
5	3.0	12024.072
6	3.5	14028.084
7	4.0	16032.096
8	4.5	18036.108
9	5.0	20040.120




Sample data points
1.0
0.5
0.0
−0.5
−1.0
y
0.0
0.2
0.4
0.6
0.8
1.0
x


Degree 1
Degree 4
Degree 16
Model
True function
Samples
y
x


x
y
z
4
6
8
10
1
2
3
4
5
6
7



	pc1	pc2	pc3
pc1	1.0	0.0	0.0
pc2	0.0	1.0	0.0
pc3	0.0	0.0	1.0

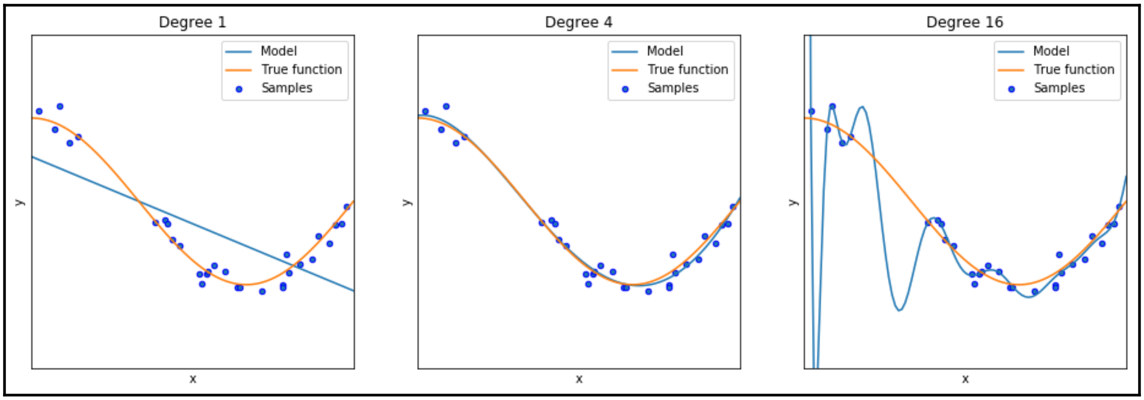
	carat	depth	table	dim_index
30066	-0.840293	1.429309	-0.205642	-0.918724
17608	0.677534	0.383359	-2.001069	0.848719
42508	-0.629484	0.034709	-0.205642	-0.568908
22842	0.719696	-0.662591	0.243215	0.908842
25957	2.553737	-1.987460	2.487499	2.147581

	carat	depth	table	dim_index
count	48537.0000	48537.0000	48537.0000	48537.0000
mean	-0.0000	-0.0000	-0.0000	0.0000
std	1.0000	1.0000	1.0000	1.0000
min	-1.2619	-13.0745	-6.4896	-1.8151
25%	-0.8403	-0.5231	-0.6545	-0.9077
50%	-0.2079	0.0347	-0.2056	-0.0236
75%	0.5089	0.5228	0.6921	0.7115
max	8.8780	12.0283	9.6692	4.4957

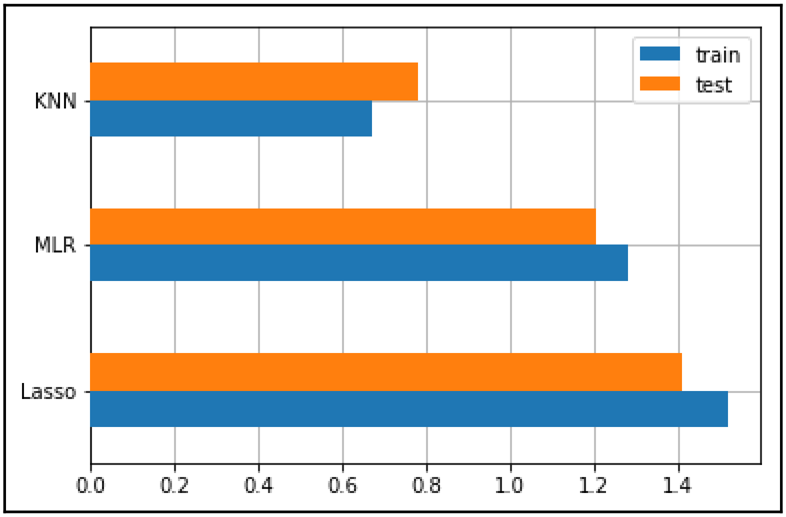
```
carat              5422.04
clarity_IF         5384.93
clarity_VVS1       5040.24
clarity_VVS2       4993.61
clarity_VS1        4616.93
clarity_VS2        4303.06
clarity_SI1        3704.82
clarity_SI2        2740.18
cut_Ideal           856.23
cut_Premium         756.77
cut_Very Good       756.17
cut_Good            609.70
table               -59.04
depth               -80.63
color_E            -217.07
color_F            -276.78
color_G            -489.66
color_H            -991.01
dim_index         -1235.23
color_I           -1480.56
color_J           -2384.35
dtype: float64
```

```
clarity_IF        5148.67
clarity_VVS1      4790.55
clarity_VVS2      4598.87
dim_index         4037.32
clarity_VS1       4006.20
clarity_VS2       3711.01
clarity_SI1       2975.99
clarity_SI2       2210.70
cut_Premium        934.69
cut_Ideal          923.11
cut_Very Good      811.81
cut_Good           607.89
depth              136.90
table               -6.19
color_E           -212.55
color_F           -361.49
color_G           -503.53
color_H           -814.98
color_I          -1111.87
color_J          -1876.45
dtype: float64
```

```
carat             4766.29
clarity_IF        1348.44
clarity_VVS2      1213.08
clarity_VVS1      1194.84
clarity_VS1        860.32
clarity_VS2        616.93
cut_Ideal          169.10
cut_Very Good       89.01
cut_Premium         55.05
clarity_SI1         33.97
cut_Good            -0.00
color_F             -0.00
color_E              0.00
table             -103.99
color_G           -124.79
depth             -145.90
color_H           -609.87
dim_index         -708.54
clarity_SI2       -768.25
color_I          -1001.55
color_J          -1780.44
dtype: float64
```



	train	test
MLR	1.28101	1.20721
Lasso	1.52062	1.40893
KNN	0.670249	0.780698

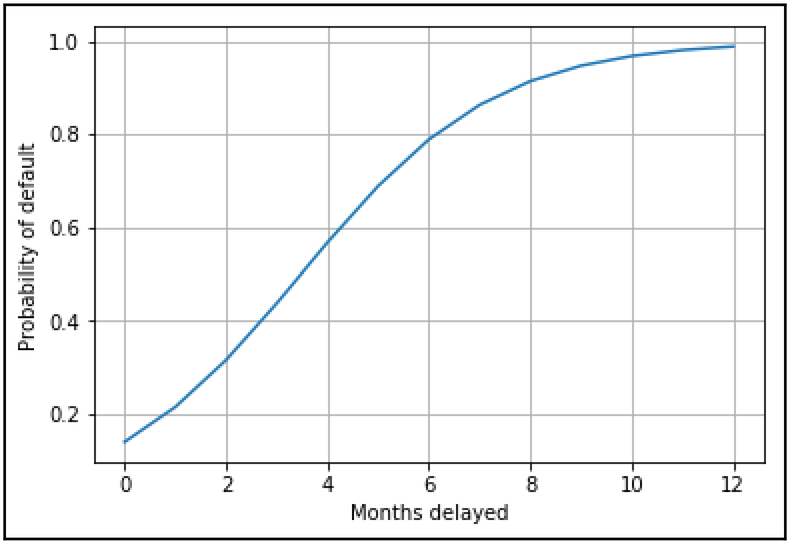


	y_true	pred_MLR	pred_Lasso	pred_KNN
8549	4434	4638.0	4993.4	4172.9
27123	17313	15503.2	14918.2	14771.3
40907	1179	1603.1	1611.9	1092.5
1375	2966	3063.0	3299.2	2915.8
41673	1240	1859.6	1567.7	978.0
35461	901	1700.1	1329.1	1158.1
30655	736	1086.4	689.6	754.6
10271	4752	6010.9	6042.6	4970.2
28928	684	904.1	762.3	753.5
26351	645	704.7	651.4	722.9

Chapter 05: Predicting Categories with Machine Learning

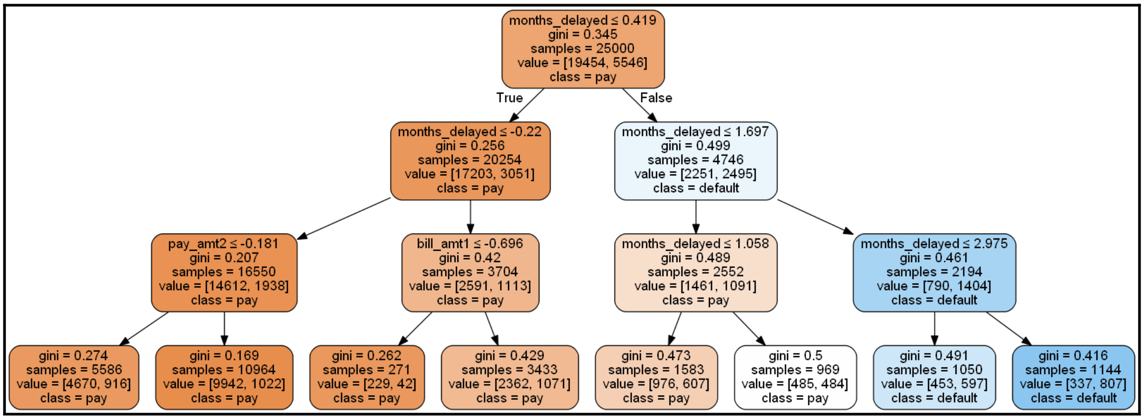
$$P(y=1|X) = \frac{1}{1+exp(-w^t X)}$$

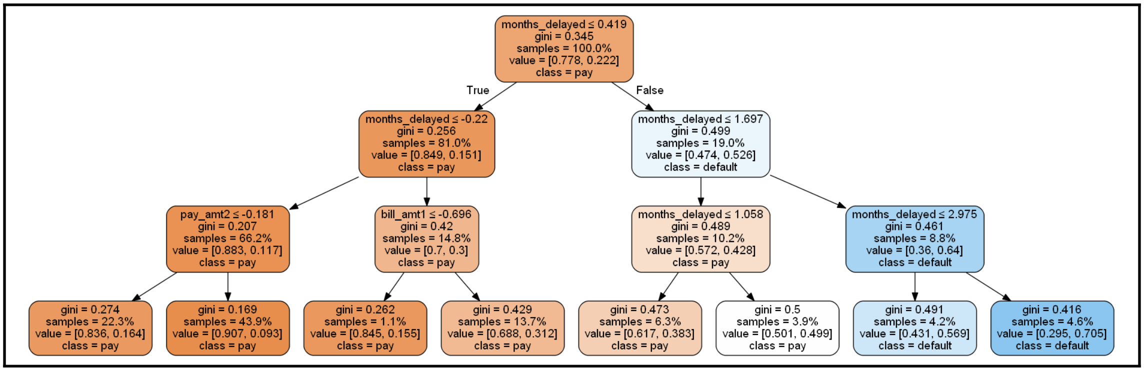
	months	pred_probs
0	0	0.139067
1	1	0.214219
2	2	0.315119
3	3	0.437107
4	4	0.567208
5	5	0.688658
6	6	0.788722
7	7	0.863022
8	8	0.914041
9	9	0.947220
10	10	0.968040
11	11	0.980813
12	12	0.988542

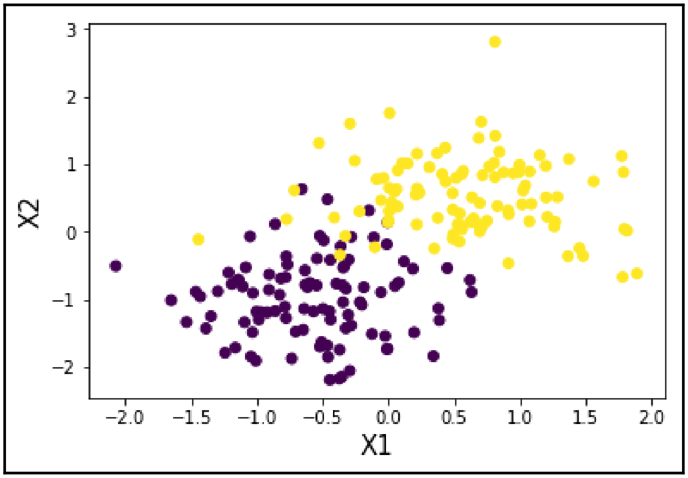


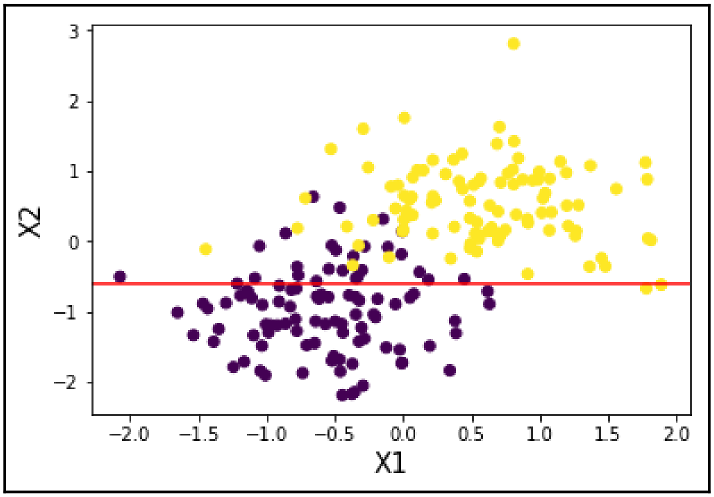

```
array([[0.80546441, 0.19453559],
       [0.89230804, 0.10769196],
       [0.80288351, 0.19711649],
       [0.85899725, 0.14100275],
       [0.19901693, 0.80098307],
       [0.82373747, 0.17626253],
       [0.70903546, 0.29096454],
       [0.79648631, 0.20351369],
       [0.81846397, 0.18153603],
       [0.73849053, 0.26150947]])
```

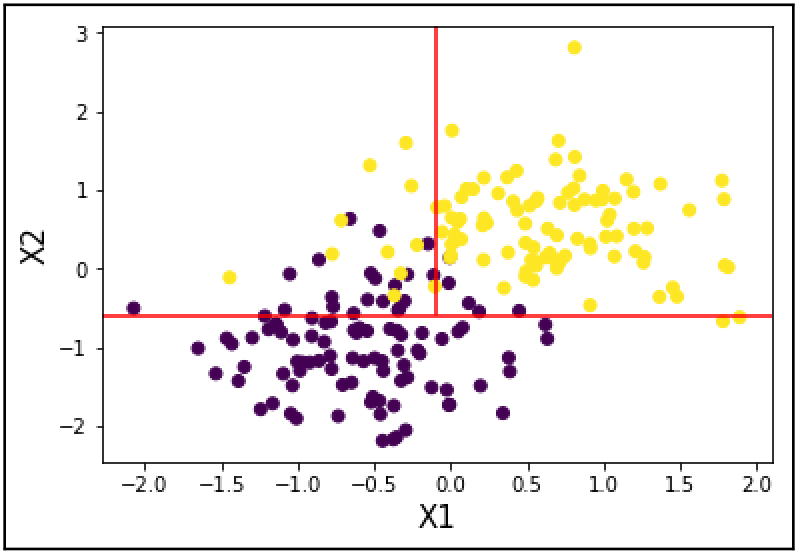
```
months_delayed    0.75
bill_amt2         0.22
bill_amt3         0.18
married           0.17
grad_school       0.13
male              0.11
university        0.11
age               0.06
pay_amt3         -0.00
pay_amt4         -0.02
bill_amt5        -0.03
pay_amt5         -0.03
pay_amt6         -0.04
bill_amt4        -0.06
bill_amt6        -0.06
bill_amt1        -0.16
limit_bal        -0.18
pay_amt1         -0.23
pay_amt2         -0.31
dtype: float64
```

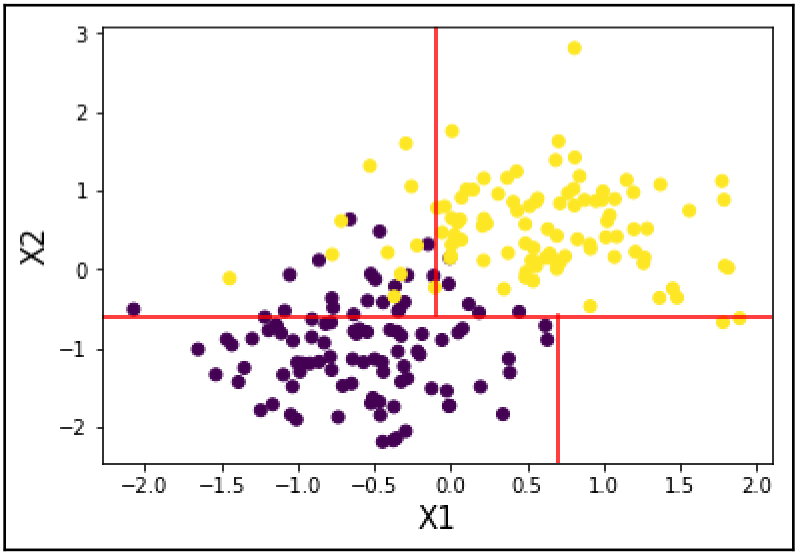



months_delayed ≤ 0.419
gini = 0.345
samples = 100.0%
value = [0.778, 0.222]
class = pay
True
False
months_delayed ≤ -0.22
gini = 0.256
samples = 81.0%
value = [0.849, 0.151]
class = pay
months_delayed ≤ 1.697
gini = 0.499
samples = 19.0%
value = [0.474, 0.526]
class = default
pay_amt2 ≤ -0.181
gini = 0.207
samples = 66.2%
value = [0.883, 0.117]
class = pay
bill_amt1 ≤ -0.696
gini = 0.42
samples = 14.8%
value = [0.7, 0.3]
class = pay
months_delayed ≤ 1.058
gini = 0.489
samples = 10.2%
value = [0.572, 0.428]
class = pay
months_delayed ≤ 2.975
gini = 0.461
samples = 8.8%
value = [0.36, 0.64]
class = default
gini = 0.274
samples = 22.3%
value = [0.836, 0.164]
class = pay
gini = 0.169
samples = 43.9%
value = [0.907, 0.093]
class = pay
gini = 0.262
samples = 1.1%
value = [0.845, 0.155]
class = pay
gini = 0.429
samples = 13.7%
value = [0.688, 0.312]
class = pay
gini = 0.473
samples = 6.3%
value = [0.617, 0.383]
class = pay
gini = 0.5
samples = 3.9%
value = [0.501, 0.499]
class = pay
gini = 0.491
samples = 4.2%
value = [0.431, 0.569]
class = default
gini = 0.416
samples = 4.6%
value = [0.295, 0.705]
class = default

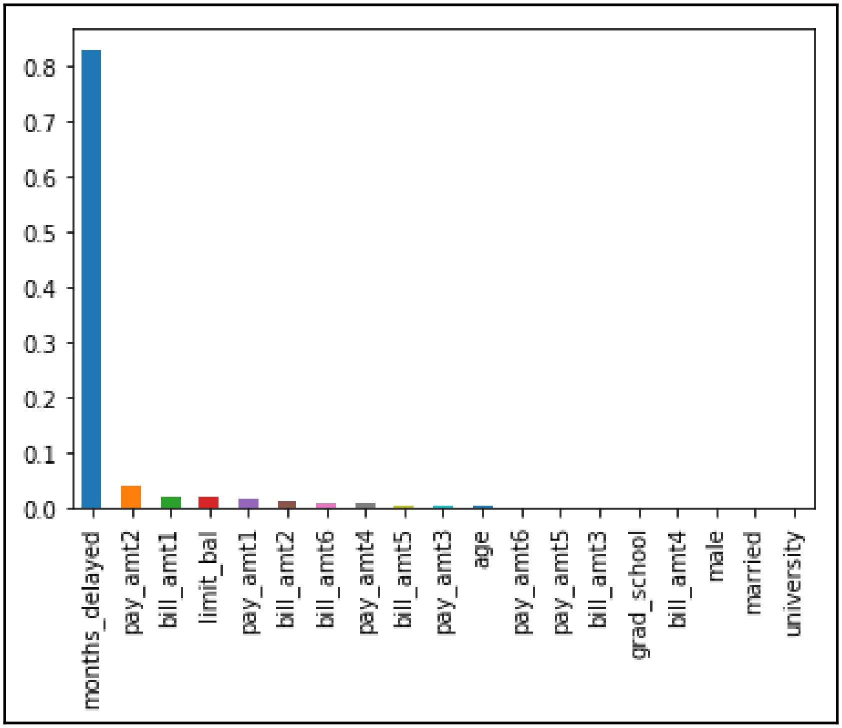

X2
X1


X2
X1


X2
X1
3
2
1
0
−1
−2
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

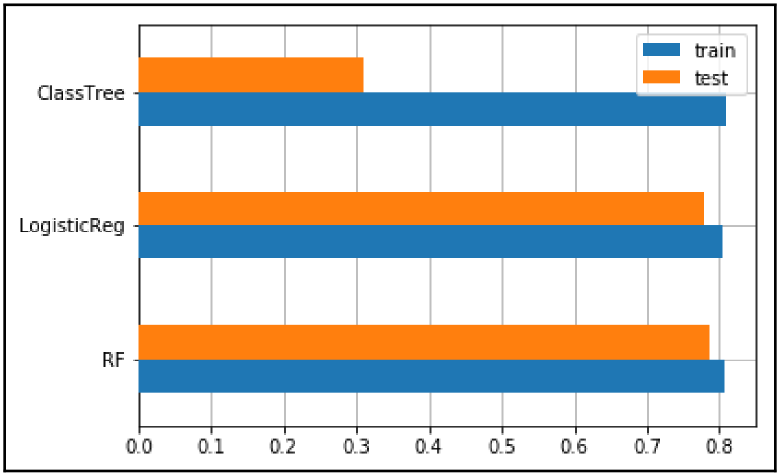

X2
X1
3
2
1
0
−1
−2
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

```
months_delayed    0.828
pay_amt2          0.042
bill_amt1         0.022
limit_bal         0.020
pay_amt1          0.018
bill_amt2         0.015
bill_amt6         0.011
pay_amt4          0.010
bill_amt5         0.007
pay_amt3          0.007
age               0.006
pay_amt6          0.004
pay_amt5          0.003
bill_amt3         0.003
grad_school       0.002
bill_amt4         0.002
male              0.000
married           0.000
university        0.000
dtype: float64
```

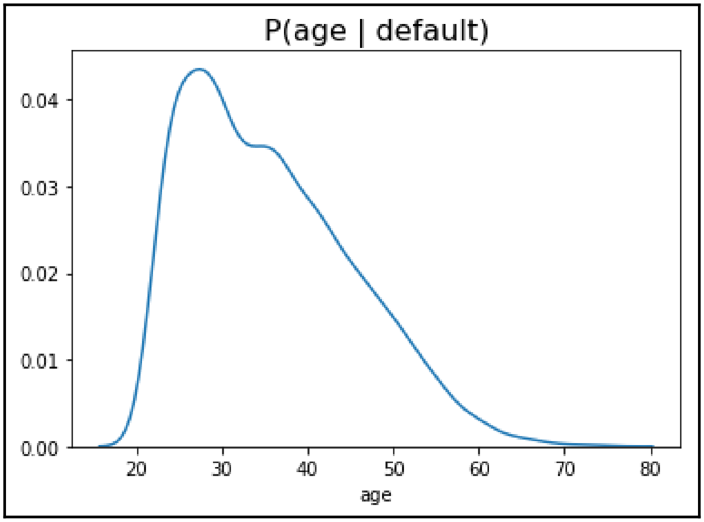

0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
months_delayed
pay_amt2
bill_amt1
limit_bal
pay_amt1
bill_amt2
bill_amt6
pay_amt4
bill_amt5
pay_amt3
age
pay_amt6
pay_amt5
bill_amt3
grad_school
bill_amt4
male
married
university

```
months_delayed    0.828
pay_amt2          0.042
bill_amt1         0.022
limit_bal         0.020
pay_amt1          0.018
bill_amt2         0.015
bill_amt6         0.011
pay_amt4          0.010
bill_amt5         0.007
pay_amt3          0.007
age               0.006
pay_amt6          0.004
pay_amt5          0.003
bill_amt3         0.003
grad_school       0.002
bill_amt4         0.002
male              0.000
married           0.000
university        0.000
dtype: float64
```

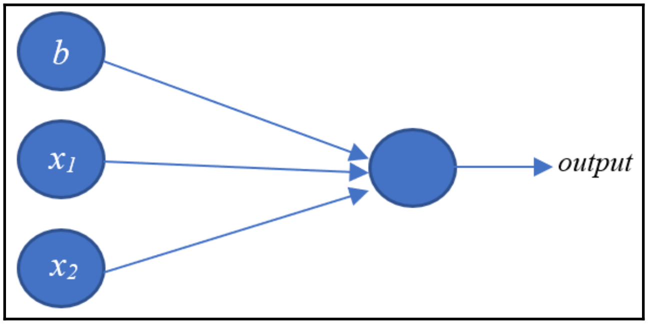
	train	test
LogisticReg	0.80372	0.7794
ClassTree	0.80824	0.3098
RF	0.80744	0.7854

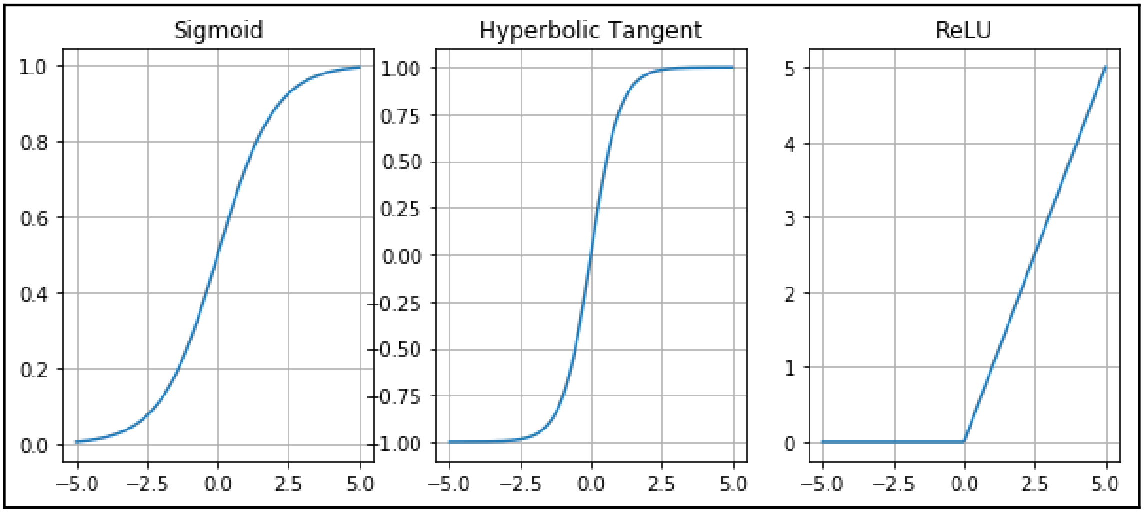


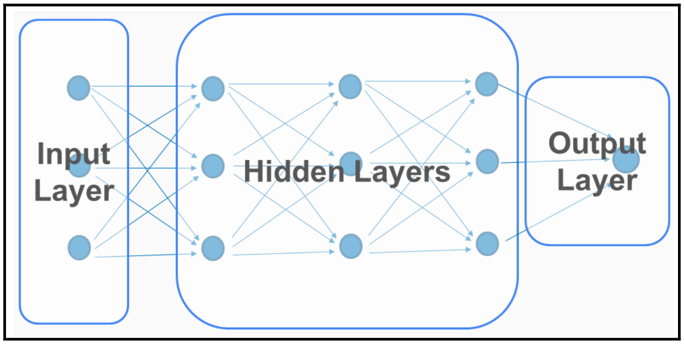
	setosa	versicolor	virginica	predicted_class
17	0.9400	0.0600	0.0000	setosa
109	0.0000	0.0332	0.9668	virginica
10	0.9512	0.0488	0.0000	setosa
78	0.0000	0.9974	0.0026	versicolor
22	0.9582	0.0418	0.0000	setosa
42	0.8494	0.1506	0.0000	setosa
148	0.0000	0.0497	0.9503	virginica
140	0.0000	0.0907	0.9093	virginica
83	0.0000	0.4270	0.5730	virginica
15	0.9959	0.0041	0.0000	setosa
133	0.0000	0.7520	0.2480	versicolor
55	0.0000	0.9998	0.0002	versicolor


P(age | default)
0.04
0.03
0.02
0.01
0.00
20
30
40
50
60
70
80
age

Chapter 06: Introducing Neural Nets for Predictive Analytics







```
array([[ 0.01396593],
       [-0.07197536],
       [-0.07281694],
       [ 0.10343548],
       [ 0.22442862]], dtype=float32)
```

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 32)                704
_________________________________________________________________
dense_2 (Dense)              (None, 16)                528
_________________________________________________________________
dense_3 (Dense)              (None, 8)                 136
_________________________________________________________________
dense_4 (Dense)              (None, 1)                 9
=================================================================
Total params: 1,377
Trainable params: 1,377
Non-trainable params: 0
```

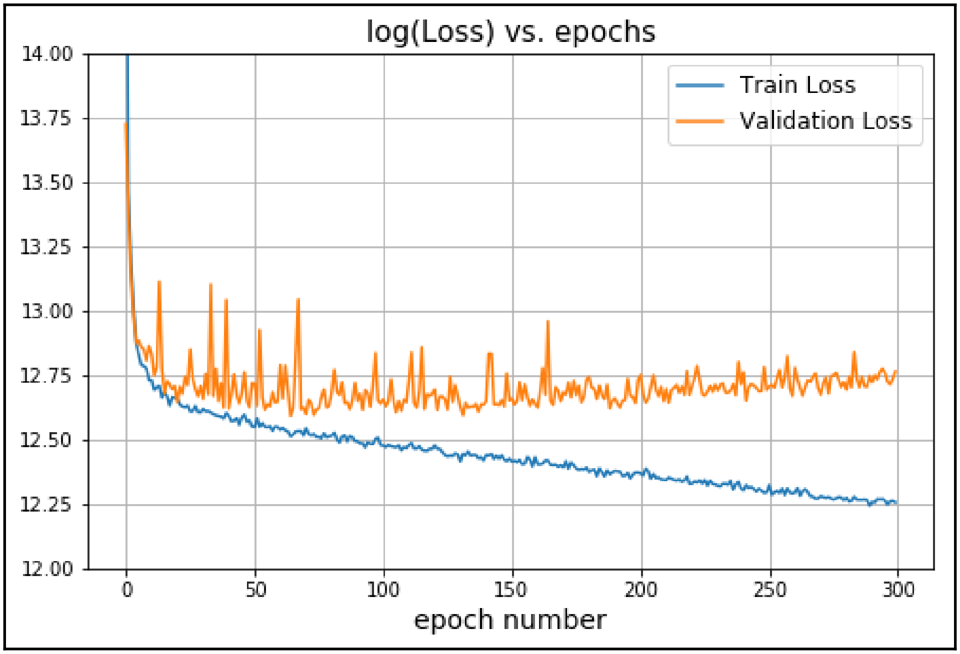
```
Epoch 1/50
48537/48537 [==============================] - 1s 17us/step - loss: 14830316.2529
Epoch 2/50
48537/48537 [==============================] - 1s 11us/step - loss: 1737586.2522
Epoch 3/50
48537/48537 [==============================] - 1s 11us/step - loss: 1222372.8833
Epoch 4/50
48537/48537 [==============================] - 1s 12us/step - loss: 1033083.3758
Epoch 5/50
48537/48537 [==============================] - 1s 12us/step - loss: 917965.2226
Epoch 6/50
48537/48537 [==============================] - 1s 12us/step - loss: 829952.7295
Epoch 7/50
48537/48537 [==============================] - 1s 12us/step - loss: 762489.6997
Epoch 8/50
48537/48537 [==============================] - 1s 11us/step - loss: 714690.7879
Epoch 9/50
```

	train	test
MLR	1.28101	1.20721
Lasso	1.52062	1.40893
KNN	0.670249	0.780698

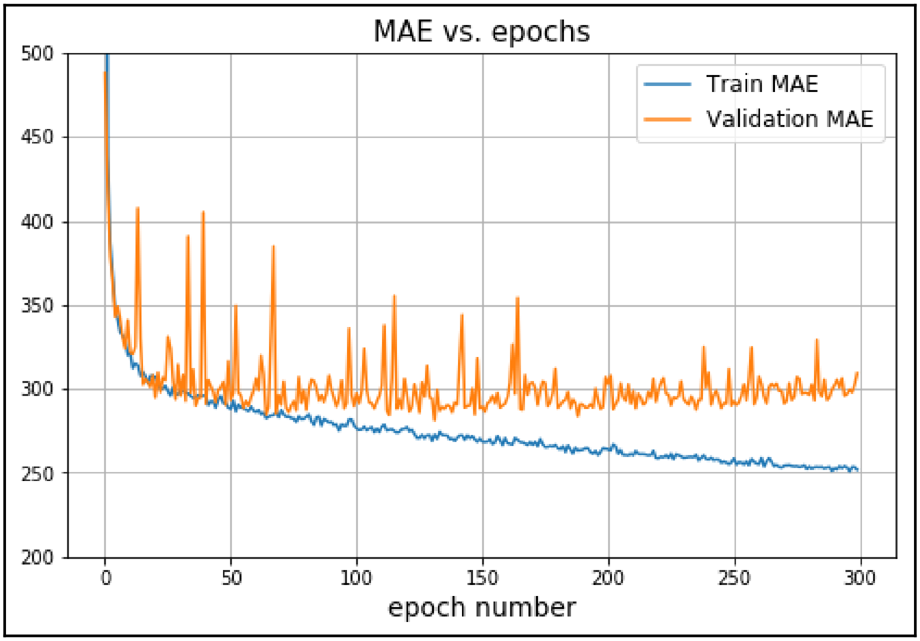
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 64)                1280
_________________________________________________________________
dense_2 (Dense)              (None, 64)                4160
_________________________________________________________________
dense_3 (Dense)              (None, 64)                4160
_________________________________________________________________
dense_4 (Dense)              (None, 64)                4160
_________________________________________________________________
dense_5 (Dense)              (None, 64)                4160
_________________________________________________________________
dense_6 (Dense)              (None, 1)                 65
=================================================================
Total params: 17,985
Trainable params: 17,985
Non-trainable params: 0
```

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_30 (Dense)             (None, 64)                1408
_________________________________________________________________
dense_31 (Dense)             (None, 64)                4160
_________________________________________________________________
dense_32 (Dense)             (None, 64)                4160
_________________________________________________________________
dense_33 (Dense)             (None, 64)                4160
_________________________________________________________________
dense_34 (Dense)             (None, 64)                4160
_________________________________________________________________
dense_35 (Dense)             (None, 64)                4160
_________________________________________________________________
dense_36 (Dense)             (None, 1)                 65
=================================================================
Total params: 22,273
Trainable params: 22,273
Non-trainable params: 0
```

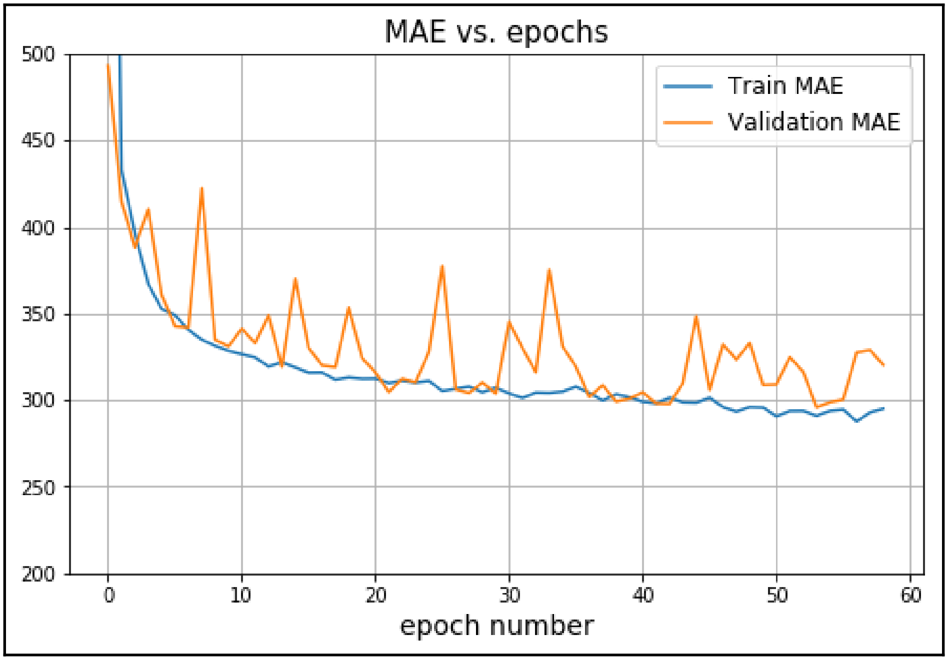
```
Train on 43683 samples, validate on 4854 samples
Epoch 1/200
43683/43683 [==============================] - 1s 34us/step - loss: 4055990.1263 - mean_squared_error: 4055990.1263 - mean_abs
olute_error: 956.1925 - val_loss: 858424.0533 - val_mean_squared_error: 858424.0533 - val_mean_absolute_error: 466.2541
Epoch 2/200
43683/43683 [==============================] - 1s 20us/step - loss: 663071.1309 - mean_squared_error: 663071.1309 - mean_absol
ute_error: 424.6258 - val_loss: 597554.1679 - val_mean_squared_error: 597554.1679 - val_mean_absolute_error: 401.7508
Epoch 3/200
43683/43683 [==============================] - 1s 19us/step - loss: 520859.1741 - mean_squared_error: 520859.1741 - mean_absol
ute_error: 382.6596 - val_loss: 507527.5170 - val_mean_squared_error: 507527.5170 - val_mean_absolute_error: 377.5142
Epoch 4/200
43683/43683 [==============================] - 1s 20us/step - loss: 436868.4289 - mean_squared_error: 436868.4289 - mean_absol
ute_error: 362.6997 - val_loss: 426952.9810 - val_mean_squared_error: 426952.9810 - val_mean_absolute_error: 354.5501
Epoch 5/200
43683/43683 [==============================] - 1s 19us/step - loss: 389967.3728 - mean_squared_error: 389967.3728 - mean_absol
ute_error: 347.0831 - val_loss: 416801.4770 - val_mean_squared_error: 416801.4770 - val_mean_absolute_error: 358.8682
```



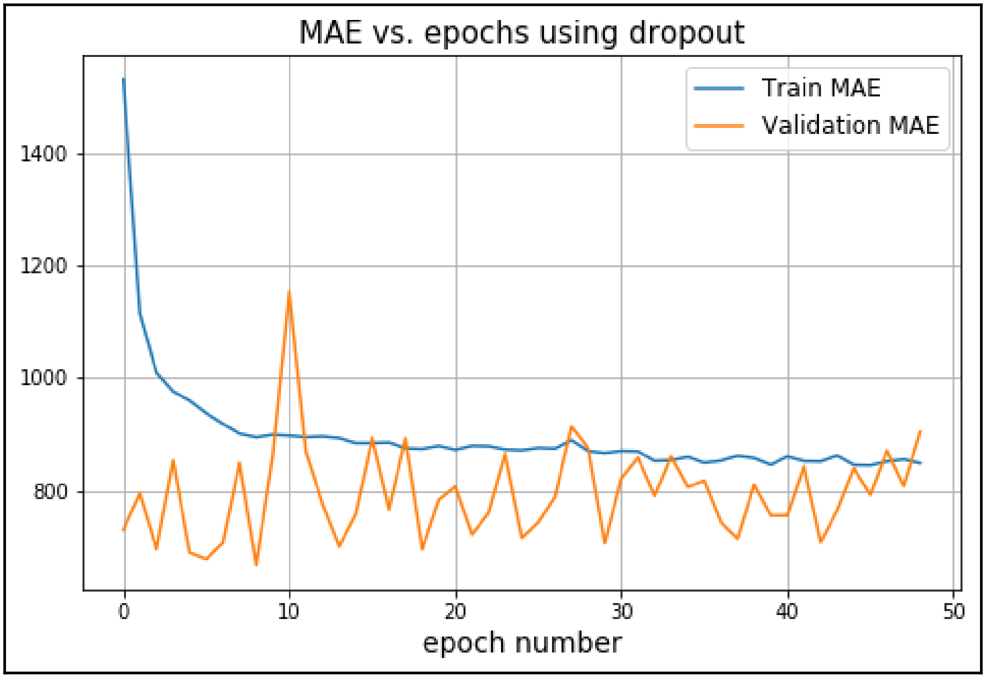
log(Loss) vs. epochs
Train Loss
Validation Loss
14.00
13.75
13.50
13.25
13.00
12.75
12.50
12.25
12.00
0
50
100
150
200
250
300
epoch number

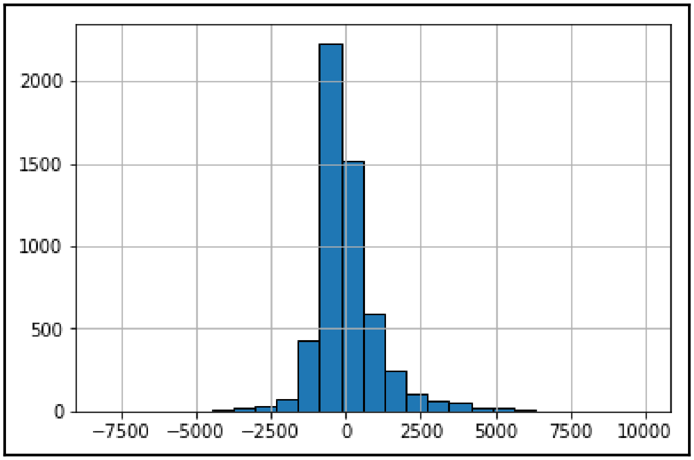


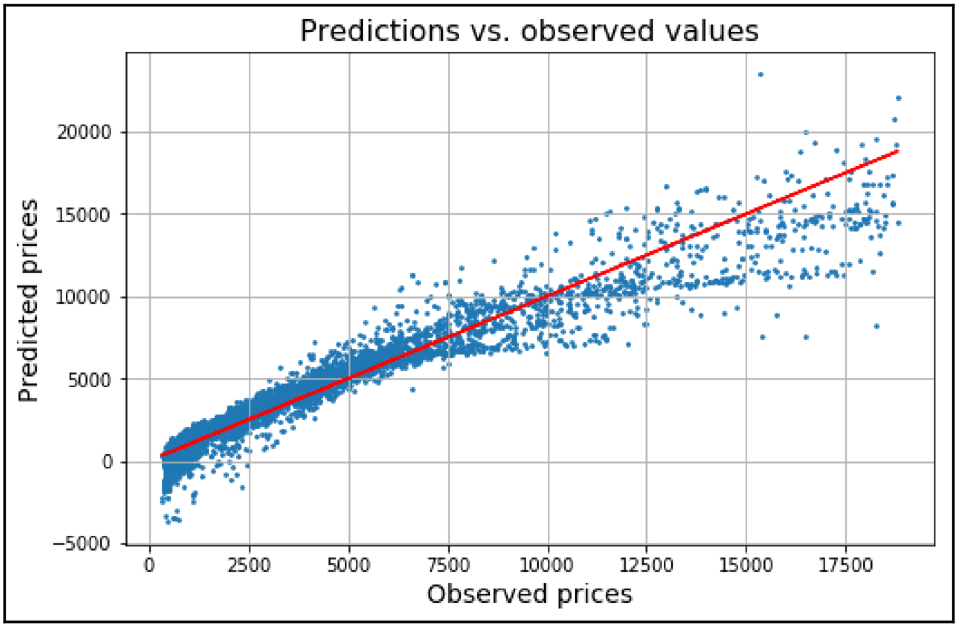
MAE vs. epochs
Train MAE
Validation MAE
500
450
400
350
300
250
200
0
50
100
150
200
250
300
epoch number


MAE vs. epochs
Train MAE
Validation MAE
500
450
400
350
300
250
200
0
10
20
30
40
50
60
epoch number

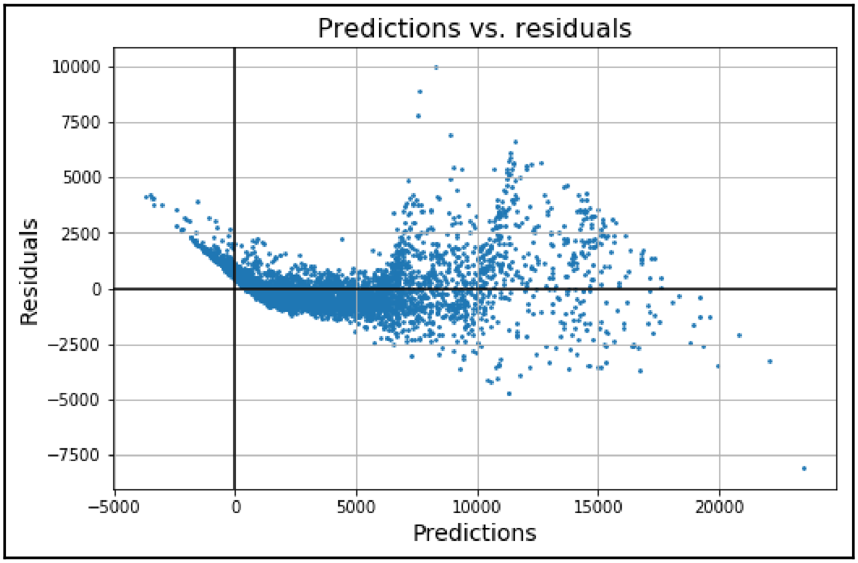
```
________________________________________________________________
Layer (type)                 Output Shape              Param #
================================================================
dropout_7 (Dropout)          (None, 21)                0
________________________________________________________________
dense_26 (Dense)             (None, 64)                1408
________________________________________________________________
dropout_8 (Dropout)          (None, 64)                0
________________________________________________________________
dense_27 (Dense)             (None, 64)                4160
________________________________________________________________
dropout_9 (Dropout)          (None, 64)                0
________________________________________________________________
dense_28 (Dense)             (None, 64)                4160
________________________________________________________________
dropout_10 (Dropout)         (None, 64)                0
________________________________________________________________
dense_29 (Dense)             (None, 64)                4160
________________________________________________________________
dropout_11 (Dropout)         (None, 64)                0
________________________________________________________________
dense_30 (Dense)             (None, 64)                4160
________________________________________________________________
dropout_12 (Dropout)         (None, 64)                0
________________________________________________________________
dense_31 (Dense)             (None, 64)                4160
________________________________________________________________
dropout_13 (Dropout)         (None, 64)                0
________________________________________________________________
dense_32 (Dense)             (None, 1)                 65
================================================================
Total params: 22,273
Trainable params: 22,273
Non-trainable params: 0
```


MAE vs. epochs using dropout
Train MAE
Validation MAE
1400
1200
1000
800
0
10
20
30
40
50
epoch number

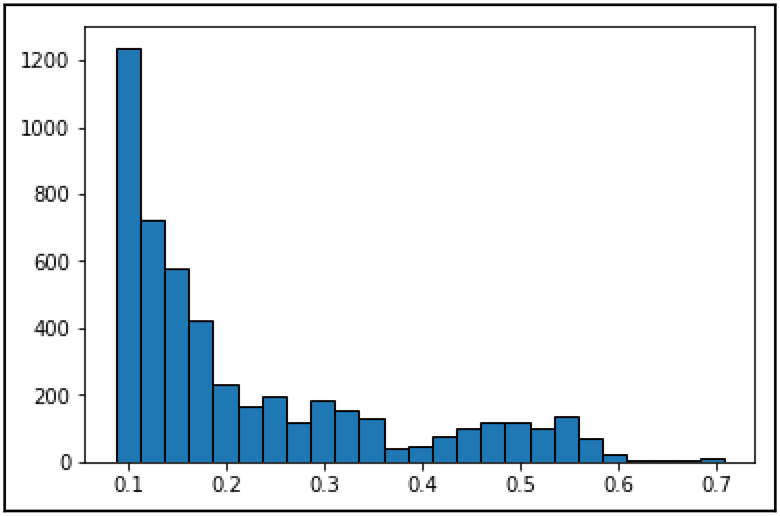

2000
1500
1000
500
0
−7500
−5000
−2500
0
2500
5000
7500
10000

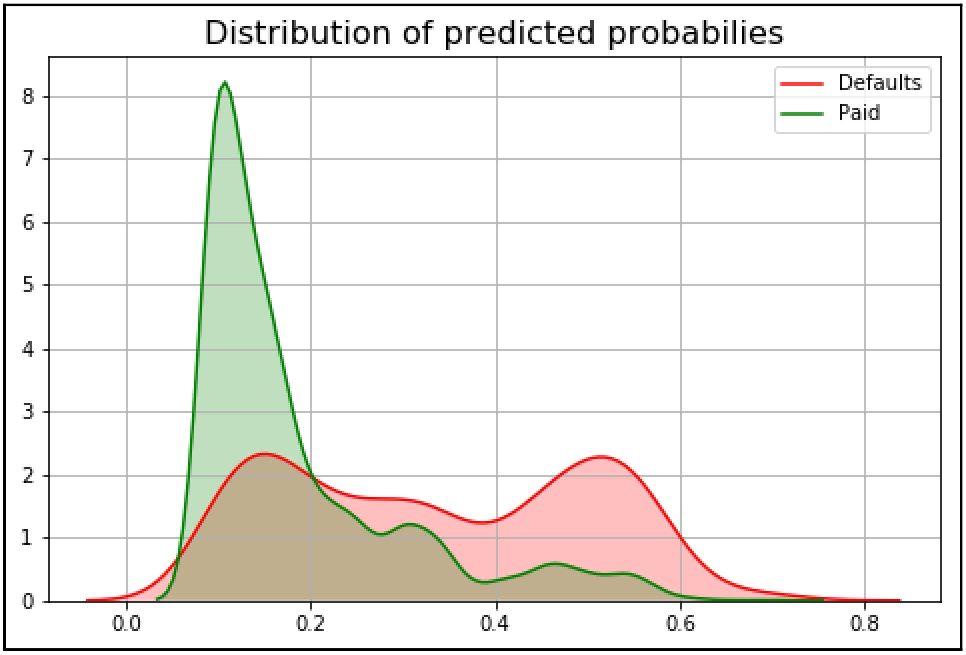

Predictions vs. observed values
Predicted prices
20000
15000
10000
5000
0
−5000
0
2500
5000
7500
10000
12500
15000
17500
Observed prices

```
39299    -312.643120
29808    -140.454585
31615    -283.365154
2714     -523.867542
5045     -569.694449
Name: y_pred, dtype: float64
```



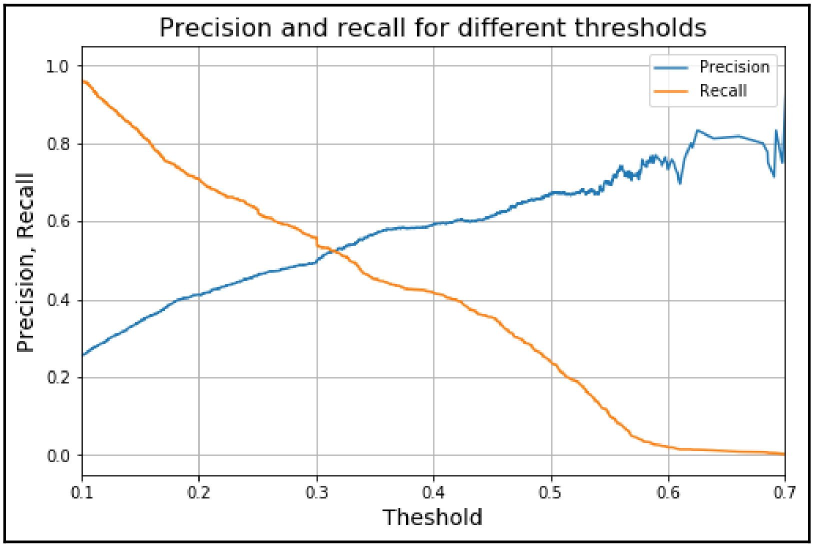
	Pred Paid	Pred Default
Obs Paid	3746	133
Obs Default	852	269

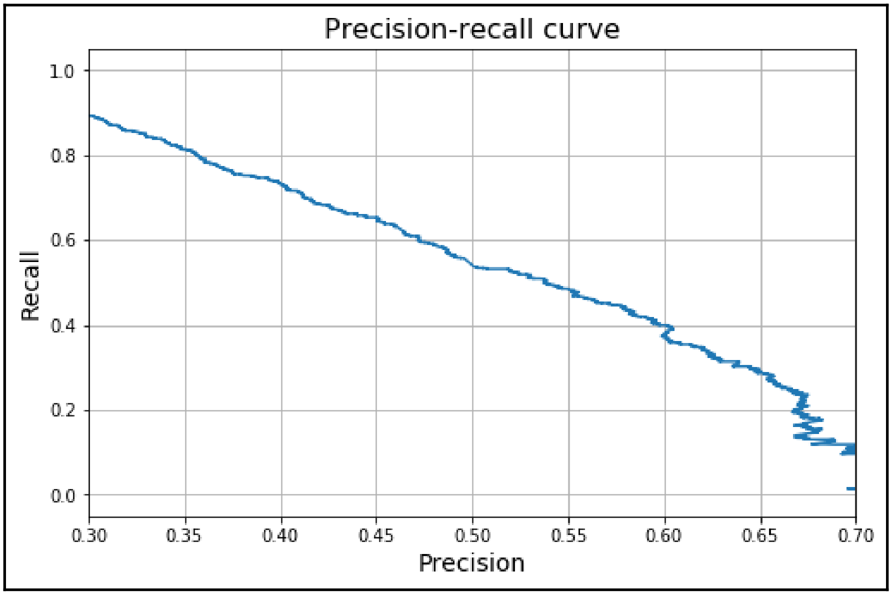


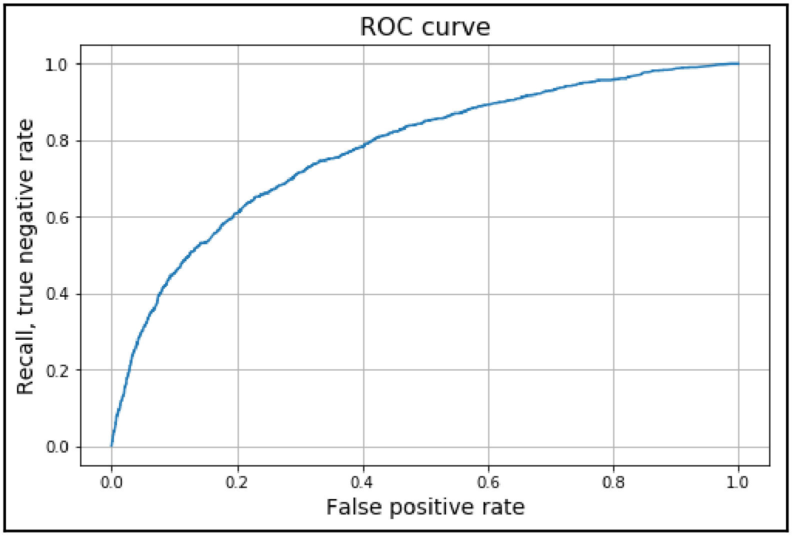


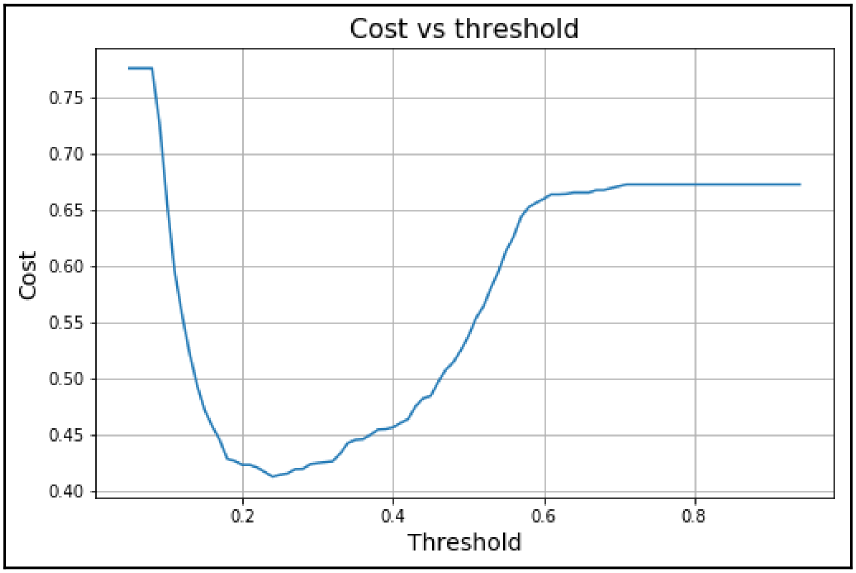
	Pred Paid	Pred Default
Obs Paid	3556	323
Obs Default	654	467

	Pred Paid	Pred Default
Obs Paid	3746	133
Obs Default	852	269



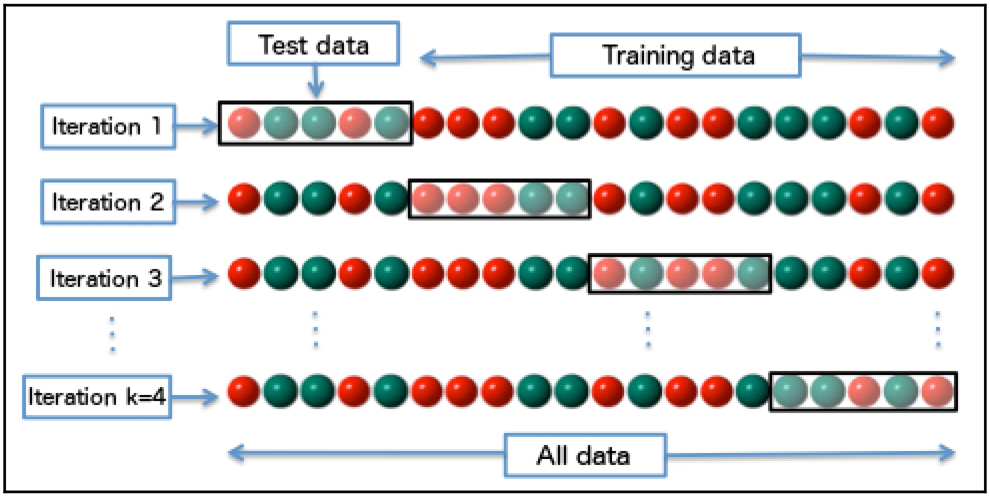



ROC curve
Recall, true negative rate
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0

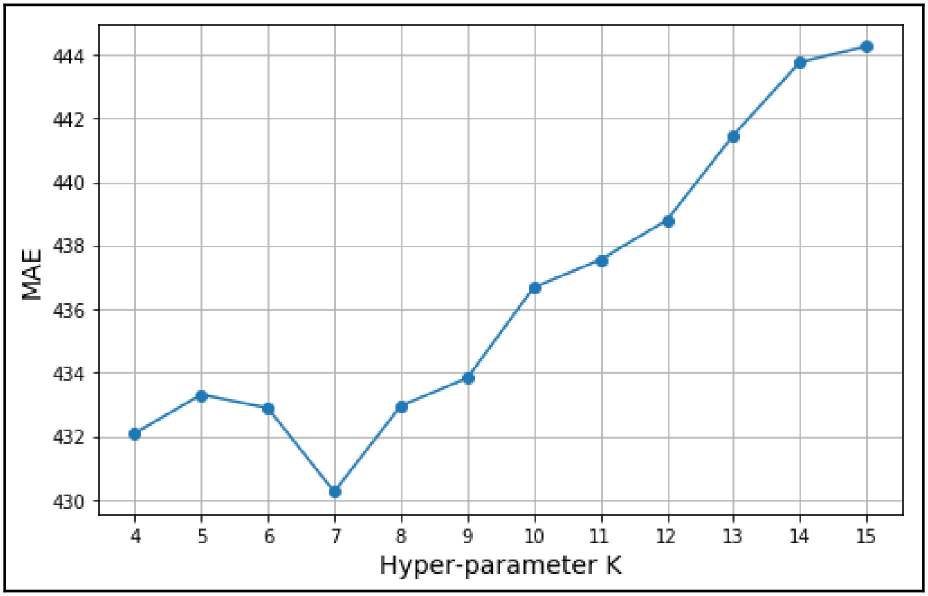

Cost vs threshold
Cost
Threshold
0.40
0.45
0.50
0.55
0.60
0.65
0.70
0.75
0.2
0.4
0.6
0.8

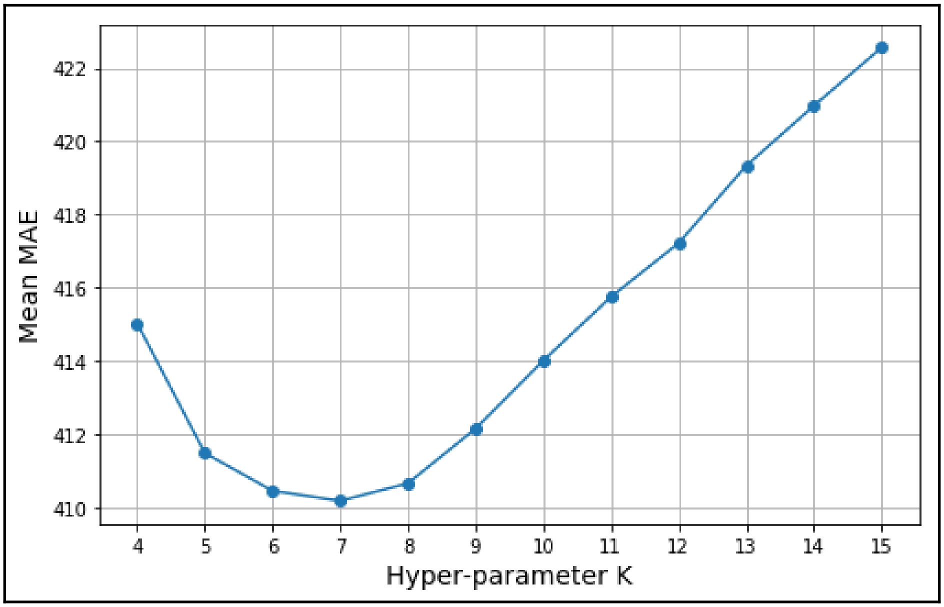
```
Precision: 45.2%, Recall: 64.9%
```

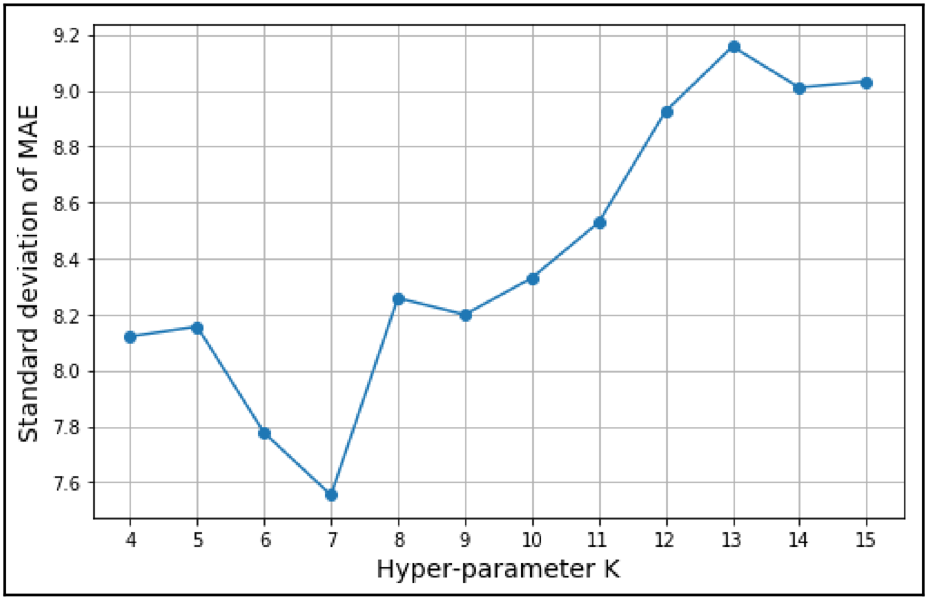
	Pred Paid	Pred Default
Obs Paid	2996	883
Obs Default	394	727



Chapter 08: Model Tuning and Improving Performance

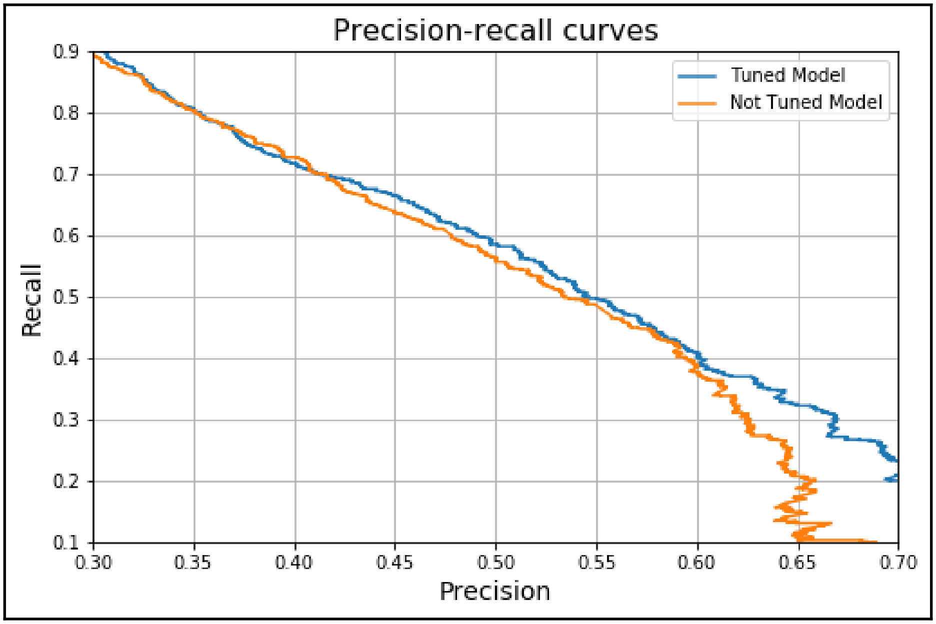


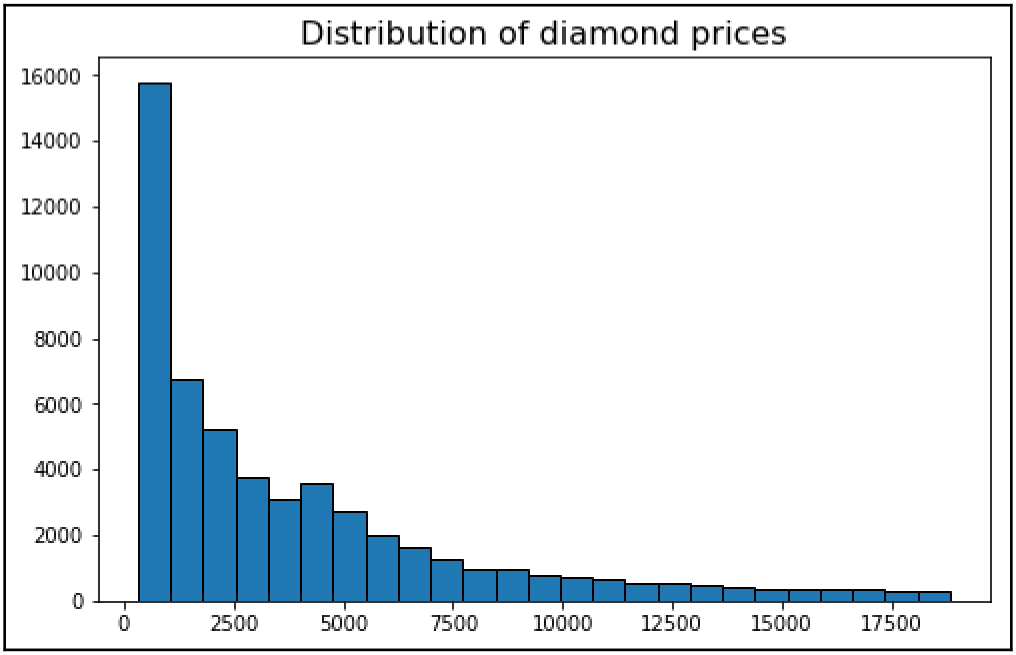


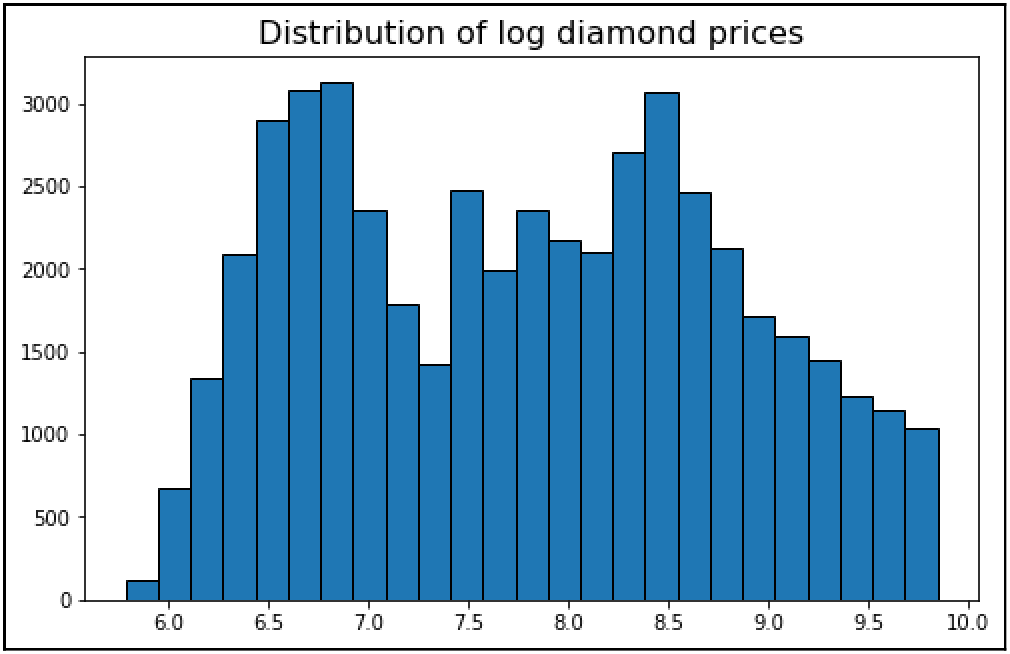


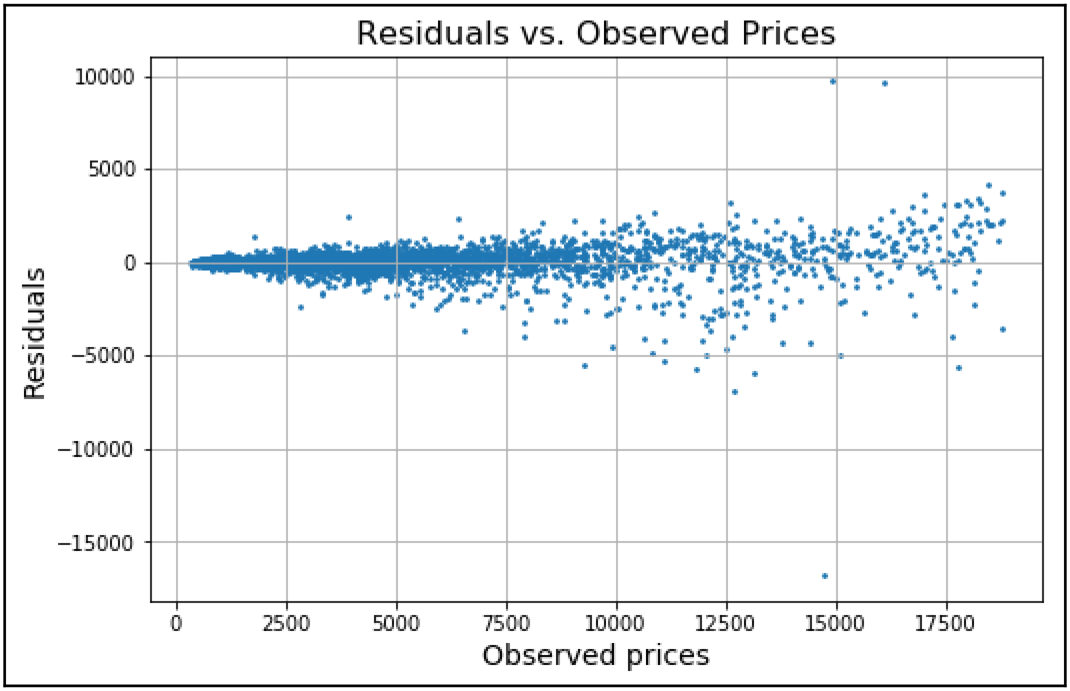
9.2
9.0
8.8
8.6
8.4
8.2
8.0
7.8
7.6
Standard deviation of MAE
4
5
6
7
8
9
10
11
12
13
14
15
Hyper-parameter K

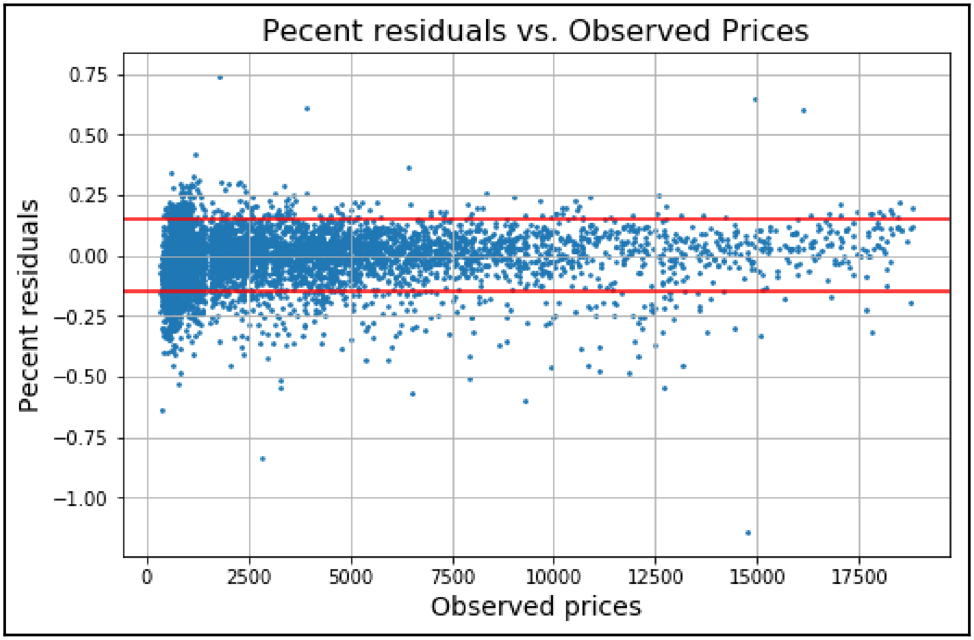
```
{'max_depth': 8, 'max_features': 10, 'n_estimators': 400}     0.771326
{'max_depth': 8, 'max_features': 10, 'n_estimators': 200}     0.771129
{'max_depth': 8, 'max_features': 10, 'n_estimators': 100}     0.770636
{'max_depth': 8, 'max_features': 19, 'n_estimators': 200}     0.770251
{'max_depth': 8, 'max_features': 19, 'n_estimators': 400}     0.770065
{'max_depth': 8, 'max_features': 19, 'n_estimators': 100}     0.769861
{'max_depth': 8, 'max_features': 10, 'n_estimators': 25}      0.768982
{'max_depth': 8, 'max_features': 4, 'n_estimators': 400}      0.768858
{'max_depth': 8, 'max_features': 4, 'n_estimators': 200}      0.768732
{'max_depth': 8, 'max_features': 4, 'n_estimators': 100}      0.768362
{'max_depth': 8, 'max_features': 19, 'n_estimators': 25}      0.767242
{'max_depth': 8, 'max_features': 4, 'n_estimators': 25}       0.765414
{'max_depth': 16, 'max_features': 4, 'n_estimators': 400}     0.765067
{'max_depth': 4, 'max_features': 19, 'n_estimators': 200}     0.764648
{'max_depth': 4, 'max_features': 19, 'n_estimators': 400}     0.764607
{'max_depth': 16, 'max_features': 4, 'n_estimators': 200}     0.764532
{'max_depth': 4, 'max_features': 10, 'n_estimators': 200}     0.764499
{'max_depth': 4, 'max_features': 10, 'n_estimators': 400}     0.764350
{'max_depth': 4, 'max_features': 19, 'n_estimators': 100}     0.764184
{'max_depth': 4, 'max_features': 10, 'n_estimators': 100}     0.763871
{'max_depth': 16, 'max_features': 10, 'n_estimators': 400}    0.762903
{'max_depth': 16, 'max_features': 10, 'n_estimators': 200}    0.762595
```

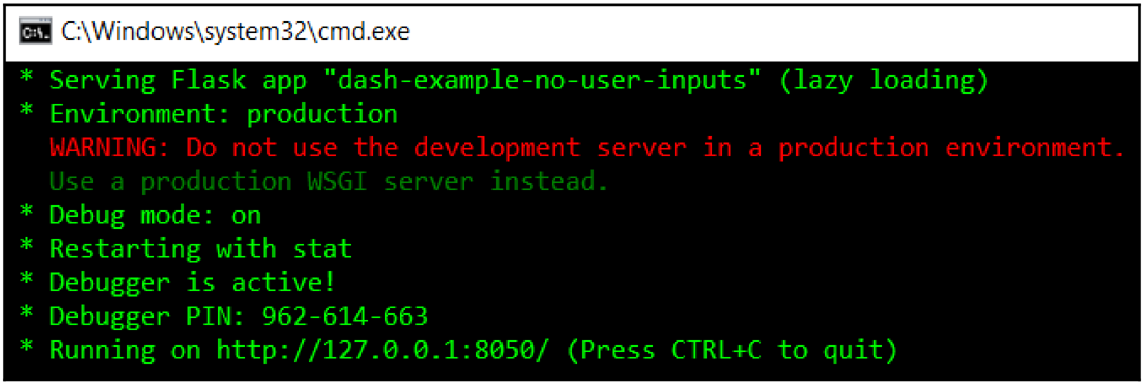

Precision-recall curves
Tuned Model
Not Tuned Model
Recall
Precision
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.30
0.35
0.40
0.45
0.50
0.55
0.60
0.65
0.70

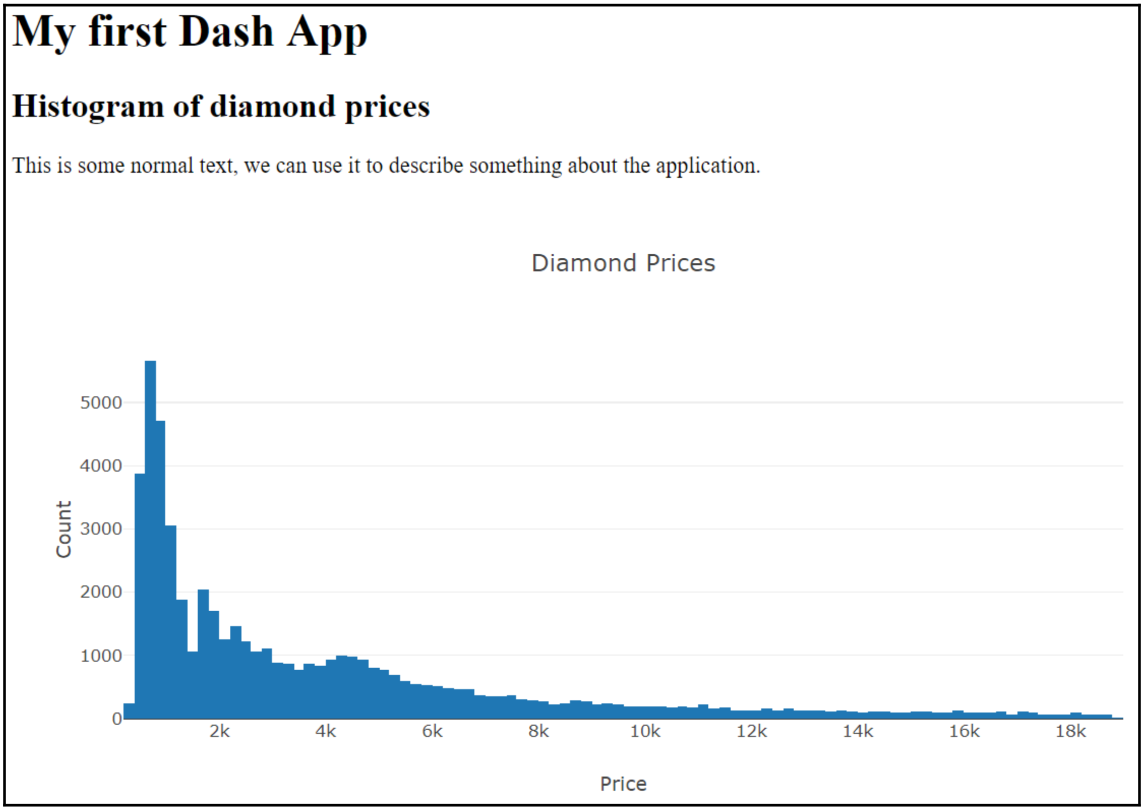

Distribution of diamond prices
16000
14000
12000
10000
8000
6000
4000
2000
0
0
2500
5000
7500
10000
12500
15000
17500

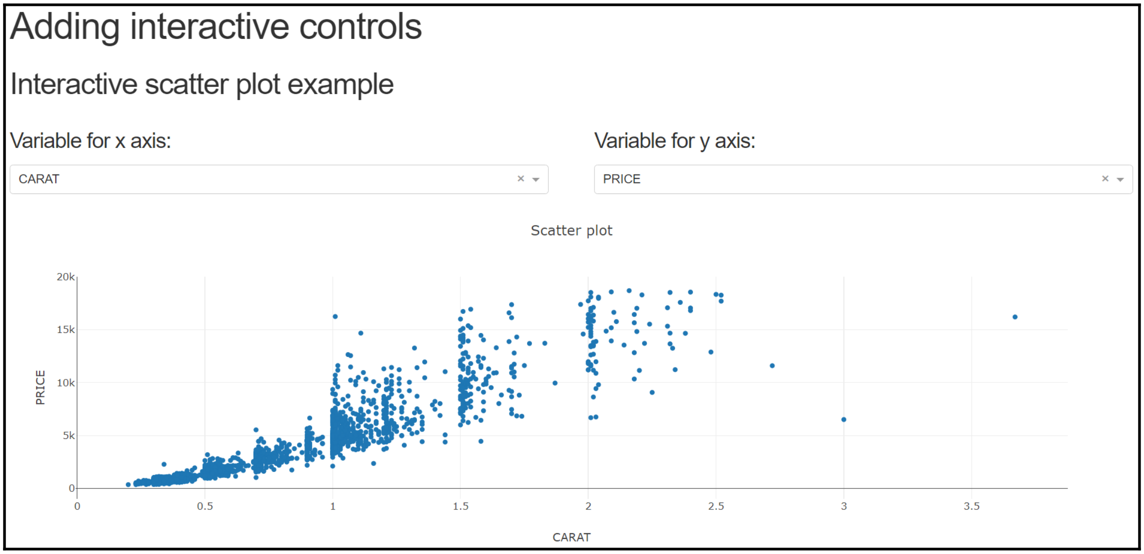

Distribution of log diamond prices
3000
2500
2000
1500
1000
500
0
6.0
6.5
7.0
7.5
8.0
8.5
9.0
9.5
10.0

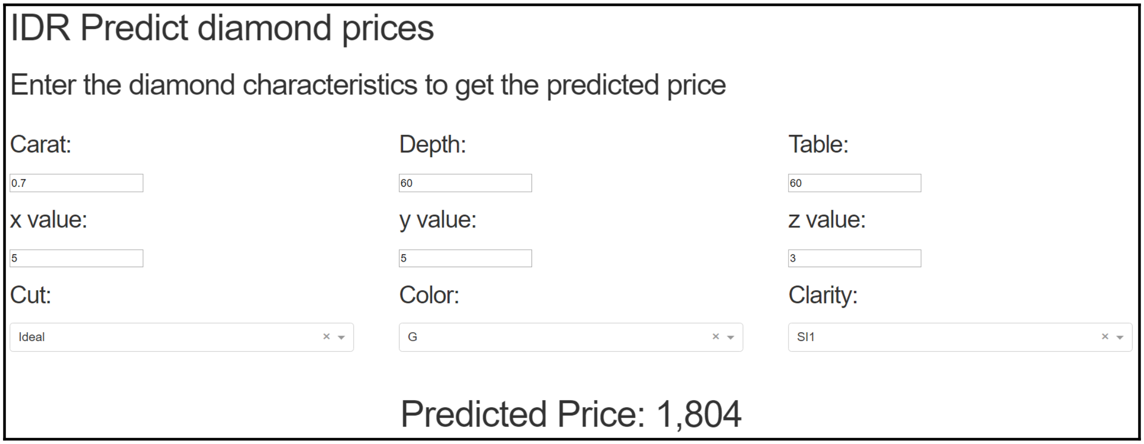

Residuals vs. Observed Prices
Residuals
10000
5000
0
−5000
−10000
−15000
0
2500
5000
7500
10000
12500
15000
17500
Observed prices


Pecent residuals vs. Observed Prices
Pecent residuals
Observed prices
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2500
5000
7500
10000
12500
15000
17500


C:\Windows\system32\cmd.exe
* Serving Flask app "dash-example-no-user-inputs" (lazy loading)
* Environment: production
WARNING: Do not use the development server in a production environment.
Use a production WSGI server instead.
* Debug mode: on
* Restarting with stat
* Debugger is active!
* Debugger PIN: 962-614-663
* Running on http://127.0.0.1:8050/ (Press CTRL+C to quit)


My first Dash App
Histogram of diamond prices
This is some normal text, we can use it to describe something about the application.
Diamond Prices
Count
5000
4000
3000
2000
1000
0
2k
4k
6k
8k
10k
12k
14k
16k
18k
Price


Adding interactive controls
Interactive scatter plot example
Variable for x axis:
CARAT
Variable for y axis:
PRICE
Scatter plot
PRICE
20k
15k
10k
5k
0
0
0.5
1
1.5
2
2.5
3
3.5
CARAT


IDR Predict diamond prices
Enter the diamond characteristics to get the predicted price
Carat:
0.7
Depth:
60
Table:
60
x value:
5
y value:
5
z value:
3
Cut:
Ideal
Color:
G
Clarity:
SI1
Predicted Price: 1,804