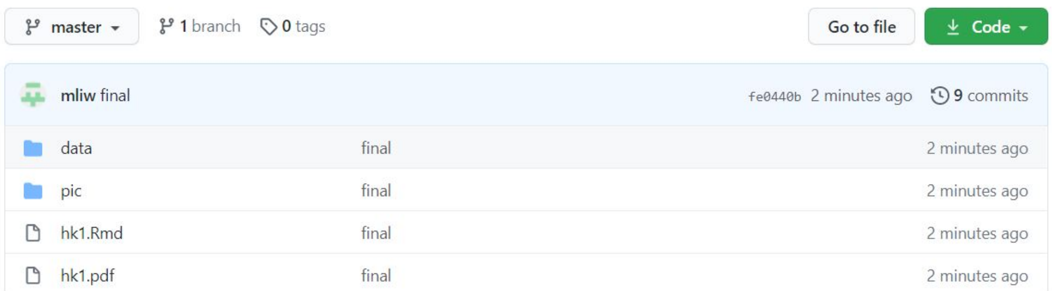
ECO395M STAT LEARNING Homework 2*

Mingwei Li, Xinyu Leng, Hongjin Long

Abstract

This document is the second homework of ECO395M STAT LEARNING.



*Mingwei Li, Xinyu Leng and Hongjin Long are master students of economics, The University of Texas at Austin

Contents

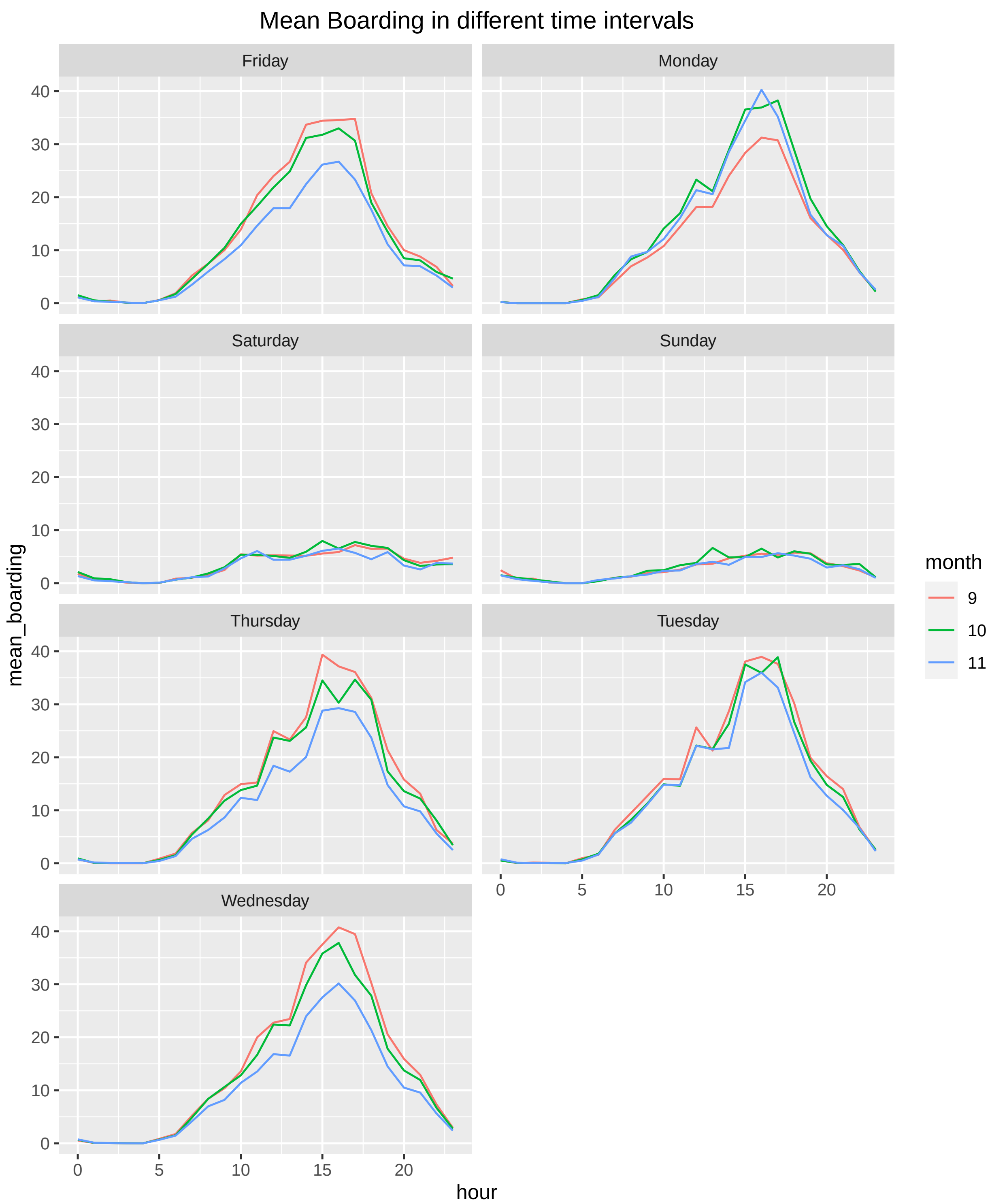
1 Problem 1: visualization

1.1 line graphs

We summarize the data and calculate average boardings

```
## # A tibble: 18 x 4
## # Groups:   hour, month [3]
##     hour month day       mean_boarding
##    <dbl> <dbl> <chr>             <dbl>
##  1     0     9 Friday            1.2
##  2     0     9 Monday            0.212
##  3     0     9 Saturday          1.82
##  4     0     9 Sunday            2.45
##  5     0     9 Thursday          0.75
##  6     0     9 Tuesday           0.562
##  7     0     9 Wednesday         0.538
##  8     0    10 Friday            1.5
##  9     0    10 Monday            0.2
## 10     0    10 Saturday          2.12
## 11     0    10 Sunday            1.51
## 12     0    10 Thursday          0.925
## 13     0    10 Tuesday           0.52
## 14     0    10 Wednesday         0.68
## 15     0    11 Friday            1.09
## 16     0    11 Monday            0.188
## 17     0    11 Saturday          1.35
## 18     0    11 Sunday            1.5
```

The faceted line plot is as follows.



(Q1_1_1) Does the hour of peak boardings change from day to day, or is it broadly similar across days?

Based on the figure, It is broadly similar across days. At about hour 15.

(Q1_1_2) Why do you think average boardings on Mondays in September look lower, compared to other days and months?

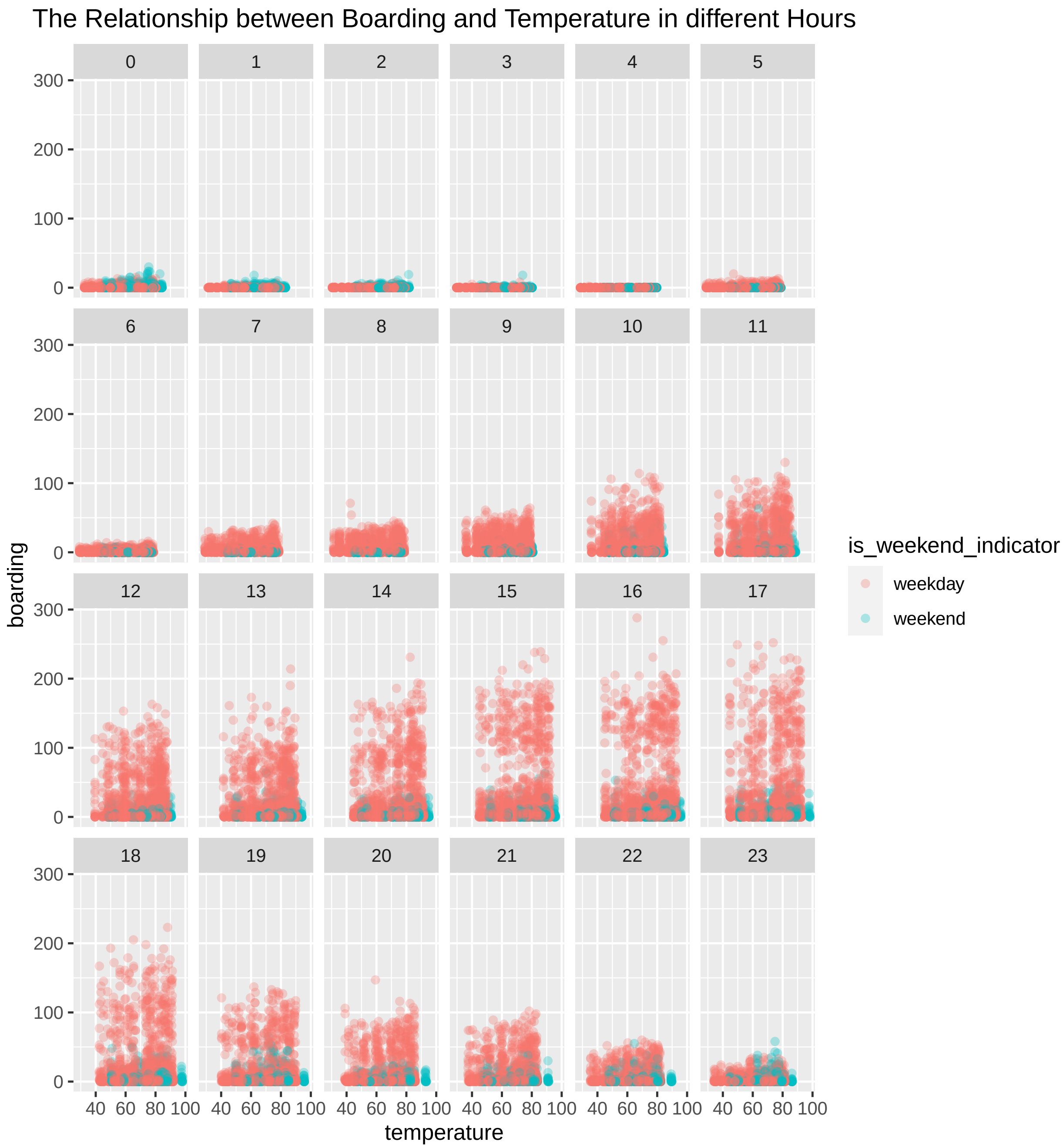
September is the beginning of Fall semester, which means there is less people on campus. On Monday, perhaps there is less courses than other days.

(Q1_1_3) Similarly, why do you think average boardings on Weds/Thurs/Fri in November look lower?

There are many midterm exams in November, which means students stay in the dorm to study for the exams without having to go outside.

1.2 scatter plots

The figure is as follows.



(Q1_2_1) When we hold hour of day and weekend status constant, does temperature seem to have a noticeable effect on the number of UT students riding the bus?

Just from the plot above, temperature doesn't have a noticeable effect on the number of UT students riding the bus.

2 Problem 2: Saratoga house prices

2.1 The Best Linear Model

The average of 5-fold corss-validation Rmse is used to evaluate a certain model.

The cross-validation Rmse of middle model is 65989.29. Our target is very simple, to find a model with cross-validation rmse lower than 65989.29. A greedy algorithm is used for feature selection, and the results are as follows.

```
##  [1] "price~livingArea"
##  [2] "69042.5521209595"
##  [3] "price~livingArea+landValue"
##  [4] "61992.1521959715"
##  [5] "price~livingArea+landValue+bathrooms"
##  [6] "60734.4870497862"
##  [7] "price~livingArea+landValue+bathrooms+waterfront"
##  [8] "59681.3922388418"
##  [9] "price~livingArea+landValue+bathrooms+waterfront+newConstruction"
## [10] "59130.6606413393"
## [11] "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating"
## [12] "58725.8085708911"
## [13] "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating+lotSize"
## [14] "58481.4628360409"
## [15] "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating+lotSize+centralAir"
## [16] "58281.8426816498"
## [17] "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating+lotSize+centralAir+age
## [18] "58149.8140719354"
## [19] "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating+lotSize+centralAir+age+
## [20] "58064.9359809818"
## [21] "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating+lotSize+centralAir+age+
## [22] "57901.1891585144"
## [23] "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating+lotSize+centralAir+age+
## [24] "57874.7183757616"
## [25] "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating+lotSize+centralAir+age+
## [26] "57850.9911286627"
## [27] "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating+lotSize+centralAir+age+
## [28] "57832.0527930649"
## [29] "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating+lotSize+centralAir+age+
## [30] "57828.0491338015"
```

```
## [1] 57828.05
```

The best variables are:

```
# "price~livingArea+landValue+bathrooms+waterfront+newConstruction+
# heating+lotSize+centralAir+age+rooms+bedrooms+fuel+pctCollege+sewer+fireplaces"
# Error:57828.05
```

Corresponding cross-validation error is 57828.05, lower than 65989.29 from the medium model.

In all, we successfully overperform the medium model!.

2.2 The Best KNN

Package kknn is used, and we slightly modify the evaluation function.

Greedy algorithm is adopted again to select the best feature combination, the results are as follows.

```
##  [1] "price~livingArea"
##  [2] "70346.039059905"
##  [3] "price~livingArea+landValue"
##  [4] "64185.1951776439"
##  [5] "price~livingArea+landValue+age"
##  [6] "60783.0749525818"
##  [7] "price~livingArea+landValue+age+pctCollege"
##  [8] "59057.3230277704"
##  [9] "price~livingArea+landValue+age+pctCollege+waterfront"
## [10] "58263.9693895441"
## [11] "price~livingArea+landValue+age+pctCollege+waterfront+newConstruction"
## [12] "58061.7316651973"
```

2.3 Analysis

The best variables and cross-validation error for KNN is

```
# "price~livingArea+landValue+age+pctCollege+waterfront+newConstruction"
# 58061.7316651973
```

The best variables and cross-validation error for linear model is

```
# "price~livingArea+landValue+bathrooms+waterfront+newConstruction+heating+
# lotSize+centralAir+age+rooms+bedrooms+fuel+pctCollege+sewer+fireplaces"
# 57828.05
```

Although the cross-validation error is lower for linear model, I still believe knn is better, as it uses only 6 variables to achieve its lowest error.

Moreover, $\frac{58061.7-57828.05}{57828.05} = 0.00404$, not very much.

3 Problem 3: Classification and retrospective sampling

Summary of the data.

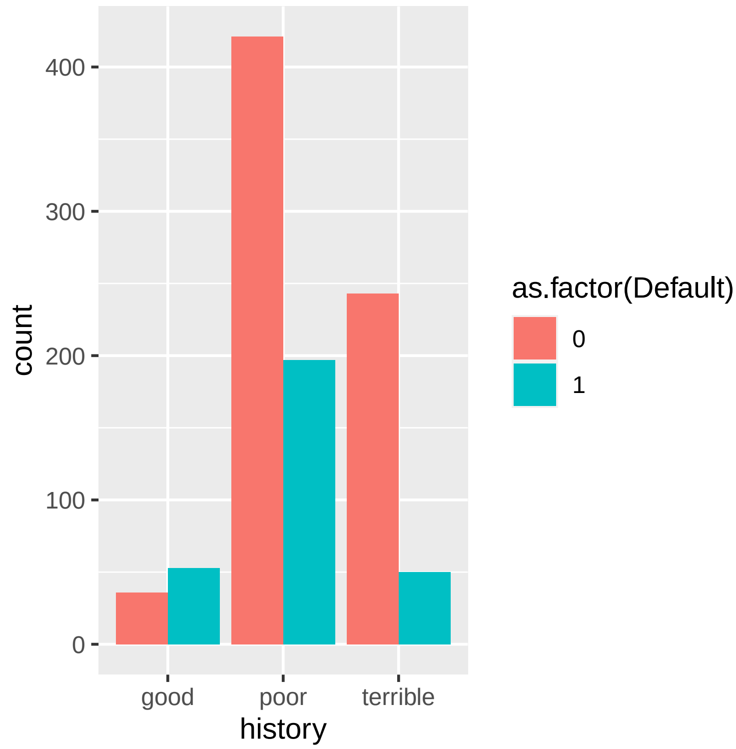
```
## Warning: package 'vcd' was built under R version 4.0.4
```

```
##        X             Default    checkingstatus1    duration       history
##  Min.   :   1.0   Min.   :0.0   A11:274         Min.   : 4.0   good    : 89
##  1st Qu.: 250.8   1st Qu.:0.0   A12:269         1st Qu.:12.0   poor    :618
##  Median : 500.5   Median :0.0   A13: 63         Median :18.0   terrible:293
##  Mean   : 500.5   Mean   :0.3   A14:394         Mean   :20.9
##  3rd Qu.: 750.2   3rd Qu.:1.0                   3rd Qu.:24.0
##  Max.   :1000.0   Max.   :1.0                   Max.   :72.0
##          purpose        amount      savings   employ     installment
##  biz         :109   Min.   :  250   A61:603   A71: 62   Min.   :1.000
##  edu         : 59   1st Qu.: 1366   A62:103   A72:172   1st Qu.:2.000
##  goods/repair:495   Median : 2320   A63: 63   A73:339   Median :3.000
##  newcar      :234   Mean   : 3271   A64: 48   A74:174   Mean   :2.973
##  usedcar     :103   3rd Qu.: 3972   A65:183   A75:253   3rd Qu.:4.000
##                     Max.   :18424                       Max.   :4.000
##  status     others      residence     property        age        otherplans
##  A91: 50   A101:907   Min.   :1.000   A121:282   Min.   :19.00   A141:139
##  A92:310   A102: 41   1st Qu.:2.000   A122:232   1st Qu.:27.00   A142: 47
##  A93:548   A103: 52   Median :3.000   A123:332   Median :33.00   A143:814
##  A94: 92              Mean   :2.845   A124:154   Mean   :35.55
##                       3rd Qu.:4.000              3rd Qu.:42.00
##                       Max.   :4.000              Max.   :75.00
##  housing        cards          job         liable        tele       foreign
##  A151:179   Min.   :1.000   A171: 22   Min.   :1.000   A191:596   foreign:963
##  A152:713   1st Qu.:1.000   A172:200   1st Qu.:1.000   A192:404   german : 37
##  A153:108   Median :1.000   A173:630   Median :1.000
##             Mean   :1.407   A174:148   Mean   :1.155
##             3rd Qu.:2.000              3rd Qu.:1.000
##             Max.   :4.000              Max.   :2.000
##    rent
##  Mode :logical
##  FALSE:821
##  TRUE :179
##
##
##
```

Count of the data.

```
##
##     good     poor terrible
##       89      618      293
```

For categories poor and terrible we see the number of nodefault are greater than the number of default. For categories good we see the number of default is greater than the number of nodefault.



Train and test split.

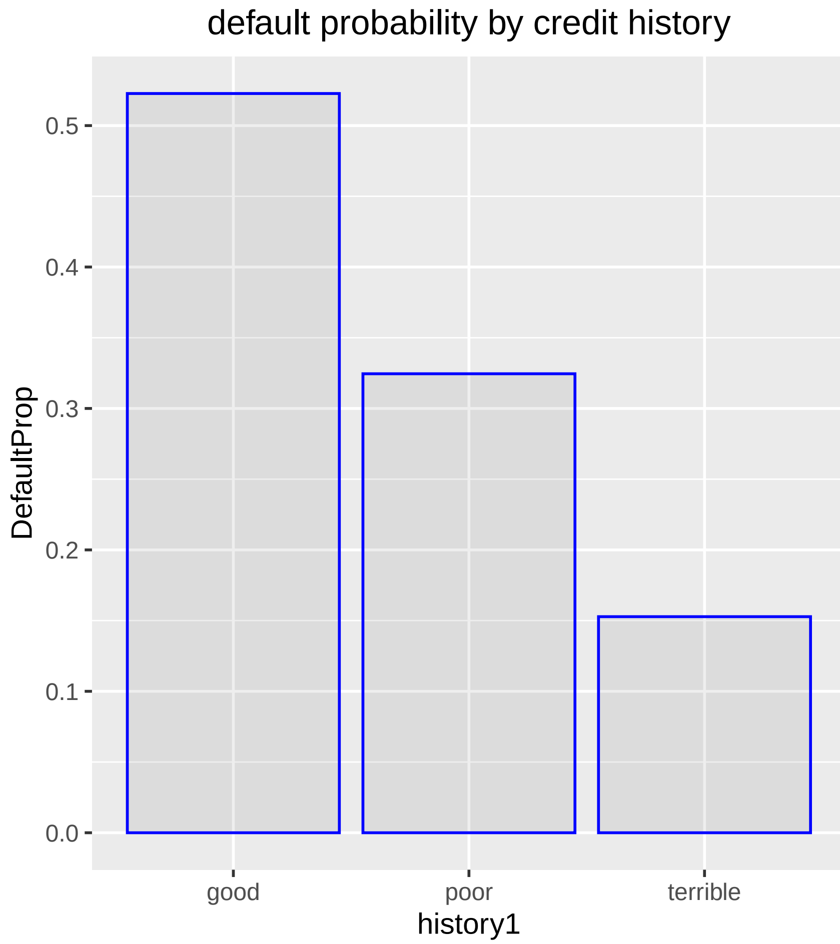
```
## [1] "train_summary"

##
##     good     poor terrible
##       62      422      216

## [1] "test_summary"

##
##     good     poor terrible
##       27      196       77
```

(Q3_1) Make a bar plot of default probability by credit history



(Q3_2) Build a logistic regression model for predicting default probability, using the variables duration + amount + installment + age + history + purpose + foreign.

The summary of model is as follows.

```
##
## Call:
## glm(formula = Default ~ duration + amount + installment + age +
##     history + purpose + foreign, family = binomial(), data = german_credittrain)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0577  -0.7934  -0.5506   0.9540   2.5422
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -2.901e-01  5.610e-01  -0.517  0.60513
## duration             1.716e-02  9.356e-03   1.834  0.06661 .
## amount               1.416e-04  4.316e-05   3.280  0.00104 **
## installment          2.105e-01  9.249e-02   2.276  0.02282 *
## age                 -1.804e-02  8.649e-03  -2.085  0.03703 *
## historypoor         -1.259e+00  2.979e-01  -4.227 2.37e-05 ***
## historyterrible     -2.169e+00  3.406e-01  -6.370 1.89e-10 ***
## purposeedu           2.726e-01  4.357e-01   0.626  0.53155
## purposegoods/repair -2.059e-01  2.901e-01  -0.710  0.47781
## purposenewcar        6.025e-01  3.162e-01   1.906  0.05670 .
## purposeusedcar      -1.223e+00  4.271e-01  -2.863  0.00419 **
## foreigngerman       -1.414e+00  7.189e-01  -1.966  0.04924 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 853.51  on 699  degrees of freedom
## Residual deviance: 730.65  on 688  degrees of freedom
## AIC: 754.65
##
## Number of Fisher Scoring iterations: 5
```

```
##         (Intercept)            duration              amount         installment
##       -0.2900604051        0.0171617189        0.0001415698        0.2105394690
##                 age         historypoor     historyterrible          purposeedu
##       -0.0180362412       -1.2591923507       -2.1693500998        0.2725939270
## purposegoods/repair       purposenewcar      purposeusedcar       foreigngerman
##       -0.2059051748        0.6025107721       -1.2229910176       -1.4136466130
```

```
##         (Intercept)            duration              amount         installment
##              -0.290               0.017               0.000               0.211
##                 age         historypoor     historyterrible          purposeedu
##              -0.018              -1.259              -2.169               0.273
## purposegoods/repair       purposenewcar      purposeusedcar       foreigngerman
##              -0.206               0.603              -1.223              -1.414
```

```
##         (Intercept)            duration              amount         installment
##           0.7482184           1.0173098           1.0001416           1.2343438
##                 age         historypoor     historyterrible          purposeedu
```

```
##           0.9821254           0.2838832           0.1142518           1.3133668
## purposegoods/repair       purposenewcar      purposeusedcar       foreigngerman
##           0.8139103           1.8266995           0.2943484           0.2432546
```

(Q3_3) What do you notice about the history variable vis-a-vis predicting defaults? What do you think is going on here?

According to the graph, the default probablitity will become higher as the borrower's credit rating is better.Because the bank matched each default with similar sets of loans that had not defaulted, including all reasonably close matches in the analysis.The sample of People with good credit is too small to lower the accuracy and they usually have fewer default samples. This resulted in a substantial oversampling of defaults

We use the confusion matrix to check out-of-sample performance

```
##       Predicted
## Actual Nodefault default
##      0       188      21
##      1        69      22
```

accuracy= (188+22)/300= 0.70

An example for predict default history by using the logistic model

```
##   X Default checkingstatus1 duration history purpose amount savings employ
## 5 5       1             A11       24    poor  newcar   4870     A61    A73
##   installment status others residence property age otherplans housing cards
## 5           3    A93   A101         4     A124  53       A143    A153     2
##    job liable tele foreign  rent
## 5 A173      2 A191 foreign FALSE
```

```
##         5
## 0.4576719
```

We could see that the defaulting probability for the 1st player in the test set is about 45.77%.

(Q3_4) In light of what you see here, do you think this data set is appropriate for building a predictive model of defaults, if the purpose of the model is to screen prospective borrowers to classify them into "high" versus "low" probability of default? Why or why not—and if not, would you recommend any changes to the bank's sampling scheme?

I think this data set is not appropriate for building a predictive model for defaults. Because the bank attempted to match each default with similar sets of loans that had not defaulted, including all reasonably close matches in the analysis. This resulted in a substantial oversampling of defaults, relative to a random sample of loans in the bank's overall portfolio.

The bank's sample size should be as large as possible. The more closer to the overall conditions, the more accurate the predictive outcome is.

4 Problem 4: Children and hotel reservations

4.1 Model building

(Q4_1)Compare the out-of-sample performance of the following models

We first load the data.

The mean f1-score from 5-fold cross-validation is applied to evaluate the performance of certain model.

1 Baseline model 1

The summary of Baseline 1(fitting on the whole data set):

```
##
## Call:
## glm(formula = as.formula(baseline_1_str), family = "binomial",
##     data = hotelsdev)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7173  -0.4921  -0.3898  -0.2163   3.4047
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -14.80395   82.46448  -0.180   0.8575
## market_segmentComplementary   12.12657   82.46459   0.147   0.8831
## market_segmentCorporate       10.04405   82.46463   0.122   0.9031
## market_segmentDirect          12.33145   82.46441   0.150   0.8811
## market_segmentGroups           9.73537   82.46459   0.118   0.9060
## market_segmentOffline_TA/TO   11.04273   82.46442   0.134   0.8935
## market_segmentOnline_TA       12.00088   82.46441   0.146   0.8843
## adults                         0.24657    0.03804   6.482 9.07e-11 ***
## customer_typeGroup            -0.31270    0.29728  -1.052   0.2929
## customer_typeTransient         0.25988    0.11091   2.343   0.0191 *
## customer_typeTransient-Party  -0.22910    0.12350  -1.855   0.0636 .
## is_repeated_guest             -0.98837    0.15055  -6.565 5.20e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 25260  on 44999  degrees of freedom
## Residual deviance: 23537  on 44988  degrees of freedom
## AIC: 23561
##
## Number of Fisher Scoring iterations: 13
```

The f1-score of fitting

```
## [1] 0
```

The 5-fold cross-validation f1-score of Baseline 1:

```
## [1] 0
```

2 Baseline model 2

The summary of Baseline 2(fitting on the whole data set):

```
##
## Call:
## glm(formula = as.formula(baseline_2_str), family = "binomial",
##     data = hotelsdev)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0395  -0.3456  -0.2291  -0.1321   3.4773
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     -1.621e+01  1.344e+02  -0.121 0.904019
## hotelResort_Hotel               -8.056e-01  5.647e-02 -14.264  < 2e-16 ***
## lead_time                        1.018e-03  2.822e-04   3.608 0.000309 ***
## stays_in_weekend_nights          6.511e-02  2.413e-02   2.699 0.006961 **
## stays_in_week_nights            -1.831e-02  1.303e-02  -1.405 0.159886
## adults                          -6.007e-01  4.479e-02 -13.412  < 2e-16 ***
## mealFB                           8.110e-01  2.669e-01   3.038 0.002379 **
## mealHB                           4.592e-02  6.787e-02   0.677 0.498696
## mealSC                          -1.205e+00  1.110e-01 -10.853  < 2e-16 ***
## mealUndefined                    2.274e-01  2.855e-01   0.796 0.425787
## market_segmentComplementary      1.218e+01  1.344e+02   0.091 0.927809
## market_segmentCorporate          1.120e+01  1.344e+02   0.083 0.933571
## market_segmentDirect             1.186e+01  1.344e+02   0.088 0.929692
## market_segmentGroups             1.089e+01  1.344e+02   0.081 0.935401
## market_segmentOffline_TA/TO      1.232e+01  1.344e+02   0.092 0.926937
## market_segmentOnline_TA          1.243e+01  1.344e+02   0.092 0.926338
## distribution_channelDirect       8.030e-01  3.313e-01   2.424 0.015367 *
## distribution_channelGDS         -1.264e+01  1.397e+02  -0.090 0.927894
## distribution_channelTA/TO        1.828e-01  2.999e-01   0.609 0.542210
## is_repeated_guest               -6.274e-01  2.117e-01  -2.964 0.003039 **
## previous_cancellations          -1.800e-01  5.438e-01  -0.331 0.740609
## previous_bookings_not_canceled  -3.881e-01  1.071e-01  -3.623 0.000291 ***
## reserved_room_typeB              1.576e+00  1.743e-01   9.037  < 2e-16 ***
## reserved_room_typeC              2.748e+00  1.767e-01  15.548  < 2e-16 ***
## reserved_room_typeD             -1.213e+00  8.089e-02 -14.996  < 2e-16 ***
## reserved_room_typeE             -4.258e-01  1.391e-01  -3.060 0.002210 **
## reserved_room_typeF              1.397e+00  1.641e-01   8.515  < 2e-16 ***
## reserved_room_typeG              2.234e+00  2.019e-01  11.065  < 2e-16 ***
## reserved_room_typeH              3.058e+00  3.761e-01   8.131 4.27e-16 ***
## reserved_room_typeL             -1.292e+01  9.831e+02  -0.013 0.989517
## assigned_room_typeB              4.182e-01  1.575e-01   2.655 0.007940 **
## assigned_room_typeC              1.704e+00  1.332e-01  12.793  < 2e-16 ***
## assigned_room_typeD              1.201e+00  7.050e-02  17.029  < 2e-16 ***
## assigned_room_typeE              1.010e+00  1.303e-01   7.752 9.01e-15 ***
## assigned_room_typeF              1.147e+00  1.603e-01   7.154 8.45e-13 ***
## assigned_room_typeG              1.284e+00  1.913e-01   6.713 1.90e-11 ***
## assigned_room_typeH              1.654e+00  3.509e-01   4.713 2.44e-06 ***
## assigned_room_typeI              1.680e+00  3.074e-01   5.464 4.66e-08 ***
## assigned_room_typeK              3.772e-01  3.712e-01   1.016 0.309619
## booking_changes                  2.418e-01  2.276e-02  10.627  < 2e-16 ***
```

```
## deposit_typeNon_Refund             2.506e-01  1.274e+00   0.197 0.844138
## deposit_typeRefundable             6.003e-01  1.037e+00   0.579 0.562680
## days_in_waiting_list              -6.104e-03  4.275e-03  -1.428 0.153294
## customer_typeGroup                -2.550e-01  3.518e-01  -0.725 0.468462
## customer_typeTransient             3.107e-01  1.224e-01   2.539 0.011113 *
## customer_typeTransient-Party      -4.431e-01  1.388e-01  -3.194 0.001405 **
## average_daily_rate                 1.046e-02  4.710e-04  22.211  < 2e-16 ***
## required_car_parking_spacesparking  1.048e-01  6.448e-02   1.626 0.103960
## total_of_special_requests          4.793e-01  2.402e-02  19.955  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 25260  on 44999  degrees of freedom
## Residual deviance: 17167  on 44951  degrees of freedom
## AIC: 17265
##
## Number of Fisher Scoring iterations: 14
```

The f1-score of fitting

```
## [1] 0.4685354
```

The 5-fold cross-validation f1-score of Baseline 2:

```
## [1] 0.4642258
```

3 Best linear model

We generate the time-stamp of year, month, day, and day of week from arrival_date.

Then, we use greedy algorithm to find the best feature combination.

```
## [1] "children~reserved_room_type"

## [1] "0.364107552305935"

## [1] "children~reserved_room_type+hotel"

## [1] "0.506343075649843"

## [1] "children~reserved_room_type+hotel+previous_cancellations"

## [1] "0.506437707712283"

## [1] "children~reserved_room_type+hotel+previous_cancellations+booking_changes"

## [1] "0.506463838208211"
```

The best feature combination is (cross-validation f1-score 0.50646)

```
## [1] "children~reserved_room_type+hotel+previous_cancellations+booking_changes"
```

The summary of best model:

```
##
## Call:
## glm(formula = as.formula(best_str), family = "binomial", data = hotelsdev)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
```

```
## -2.2540  -0.3590  -0.3169  -0.2191   3.0213
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -2.96639    0.03032 -97.844  < 2e-16 ***
## reserved_room_typeB     1.87374    0.11061  16.940  < 2e-16 ***
## reserved_room_typeC     4.44067    0.12399  35.814  < 2e-16 ***
## reserved_room_typeD     0.28068    0.05773   4.862 1.16e-06 ***
## reserved_room_typeE     1.02964    0.07676  13.414  < 2e-16 ***
## reserved_room_typeF     3.11237    0.06710  46.384  < 2e-16 ***
## reserved_room_typeG     3.93134    0.08360  47.028  < 2e-16 ***
## reserved_room_typeH     4.64365    0.16551  28.057  < 2e-16 ***
## reserved_room_typeL    -6.84816   84.47668  -0.081  0.93539
## hotelResort_Hotel      -0.75148    0.04788 -15.696  < 2e-16 ***
## previous_cancellations -1.88730    0.68029  -2.774  0.00553 **
## booking_changes         0.25691    0.02015  12.747  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 25260  on 44999  degrees of freedom
## Residual deviance: 19702  on 44988  degrees of freedom
## AIC: 19726
##
## Number of Fisher Scoring iterations: 9
```

The f1-score of fitting

```
## [1] 0.5067889
```

The 5-fold cross-validation f1-score of best model:

```
## [1] 0.5064638
```

4 Analysis

The cross-validation f1-scores of 3 models are as follows:

Baseline1 model 1: 0

Baseline1 model 2: 0.4642258

Best model: 0.5064638

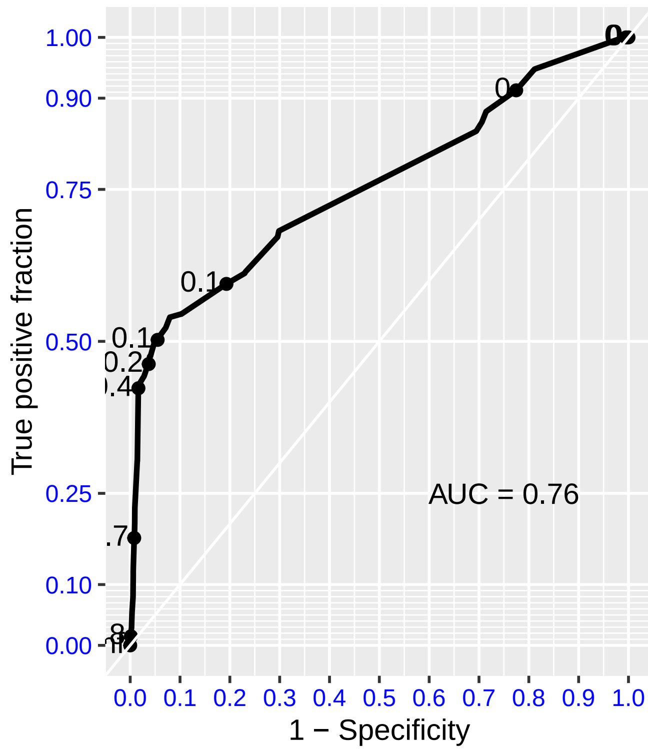
The best model is the best model with the highest f1-score.

4.2 Model validation: step 1

(Q4_2)Produce an ROC curve for your best model, using the data in hotels_val: that is, plot TPR(t) versus FPR(t) as you vary the classification threshold t.

We first fit on the hotelsdev data. Then we load hotels_val to conduct validation and draw ROC graph.

The plot is as follows:



4.3 Model validation: step 2

(Q4_3)How well does your model do at predicting the total number of bookings with children in a group of 250 bookings? Summarize this performance across all 20 folds of the val set in an appropriate figure or table.

We first fit on the hotelsdev data. Then we load hotels_val to calculate prediction accuracy.

The hotels_val is divided into 20 folds.

The following is the summary of expected number of bookings with children for that fold and actual number for that fold.

```
##    actual_num predict_num difference
## 1          18          19          1
## 2          20          22          2
## 3          12          17          5
## 4          17          17          0
## 5          23          19         -4
## 6          25          23         -2
## 7          28          21         -7
## 8          17          17          0
## 9          24          23         -1
## 10         18          21          3
## 11         14          21          7
## 12         19          22          3
## 13         17          22          5
## 14         18          22          4
## 15         19          19          0
## 16         20          18         -2
## 17         24          23         -1
## 18         24          24          0
## 19         21          21          0
## 20         24          21         -3
```

The following figure demonstrates the distribution of difference beween actual number and expected number among 20 folds.