

ECO395M STAT LEARNING Homework 4*

Mingwei Li, Xinyu Leng, Hongjin Long

Abstract

This document is the fourth homework of ECO395M STAT LEARNING.


👤 master ▾

🔗 1 branch

🏷 0 tags

Go to file

📄 Code ▾

 mliw final

fe0440b 2 minutes ago ⌚ 9 commits

📁 data	final	2 minutes ago
📁 pic	final	2 minutes ago
📄 hk1.Rmd	final	2 minutes ago
📄 hk1.pdf	final	2 minutes ago

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

Contents

1	Clustering and PCA	3
1.1	Benchmark Model	4
1.2	PCA Model	6
1.3	Kmeans Model	9
1.4	Results and Conclusions	11
2	Market segmentation	12
3	Association rules for grocery purchases	13
4	Author attribution	14
4.1	Part 1: Preprocess data	14
4.2	Part 2:Model evaluation	15

1 Clustering and PCA

We have 6497 observations.

```
## [1] 6497 13
```

We would discuss 3 cases:

- (1) Benchmark model: We fit on the original wine.csv data. logistic model is used to predict color and linear model is used to predict quality. f1-score and rmse are used to measure the quality of fitting.
- (2) PCA model V.S Benchmark model: unsupervised technique(PCA) reduce 11 features to only 1 feature(PC_1). We would use the single feature PC_1 to predict color(with logistic model) and to predict quality(with linear model).
- (3) kmeans model V.S Benchmark model: unsupervised technique(k-means) reduce 11 features to only 1 feature($class$). We would use the single feature $class$ to predict color(with logistic model) and to predict quality(with linear model).

1.1 Benchmark Model

Details of benchmark model for color:

```
##
## Call:
## glm(formula = color ~ . - color - quality, family = binomial(),
##      data = wine)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -6.6815  -0.0558  -0.0184  -0.0011   5.7002
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -1.842e+03  1.857e+02  -9.919  < 2e-16 ***
## fixed.acidity    -3.849e-01  2.345e-01  -1.641   0.1008
## volatile.acidity  6.227e+00  1.020e+00   6.106 1.02e-09 ***
## citric.acid      -2.603e+00  1.169e+00  -2.226   0.0260 *
## residual.sugar   -9.453e-01  1.008e-01  -9.378  < 2e-16 ***
## chlorides        2.225e+01  3.983e+00   5.585 2.33e-08 ***
## free.sulfur.dioxide 6.694e-02  1.333e-02   5.023 5.08e-07 ***
## total.sulfur.dioxide -5.323e-02  4.896e-03 -10.872  < 2e-16 ***
## density          1.841e+03  1.894e+02   9.720  < 2e-16 ***
## pH              -1.718e+00  1.415e+00  -1.214   0.2246
## sulphates        3.099e+00  1.244e+00   2.491   0.0128 *
## alcohol          1.900e+00  2.774e-01   6.850 7.40e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 7250.98  on 6496  degrees of freedom
## Residual deviance:  428.87  on 6485  degrees of freedom
## AIC: 452.87
##
## Number of Fisher Scoring iterations: 9
```

The f1-score is:

```
## [1] 0.9896714
```

Details of benchmark model for quality:

```
##
## Call:
## lm(formula = quality ~ . - color - quality, data = wine)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.7569  -0.4597  -0.0412   0.4694   2.9907
##
## Coefficients:
```

```

##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.576e+01  1.189e+01   4.688 2.81e-06 ***
## fixed.acidity      6.768e-02  1.557e-02   4.346 1.41e-05 ***
## volatile.acidity  -1.328e+00  7.737e-02 -17.162 < 2e-16 ***
## citric.acid       -1.097e-01  7.962e-02  -1.377   0.168
## residual.sugar     4.356e-02  5.156e-03   8.449 < 2e-16 ***
## chlorides         -4.837e-01  3.327e-01  -1.454   0.146
## free.sulfur.dioxide 5.970e-03  7.511e-04   7.948 2.22e-15 ***
## total.sulfur.dioxide -2.481e-03  2.767e-04  -8.969 < 2e-16 ***
## density          -5.497e+01  1.214e+01  -4.529 6.04e-06 ***
## pH                4.393e-01  9.037e-02   4.861 1.20e-06 ***
## sulphates         7.683e-01  7.612e-02  10.092 < 2e-16 ***
## alcohol           2.670e-01  1.673e-02  15.963 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7353 on 6485 degrees of freedom
## Multiple R-squared:  0.2921, Adjusted R-squared:  0.2909
## F-statistic: 243.3 on 11 and 6485 DF,  p-value: < 2.2e-16

```

The rmse is:

```
## [1] 0.7346533
```

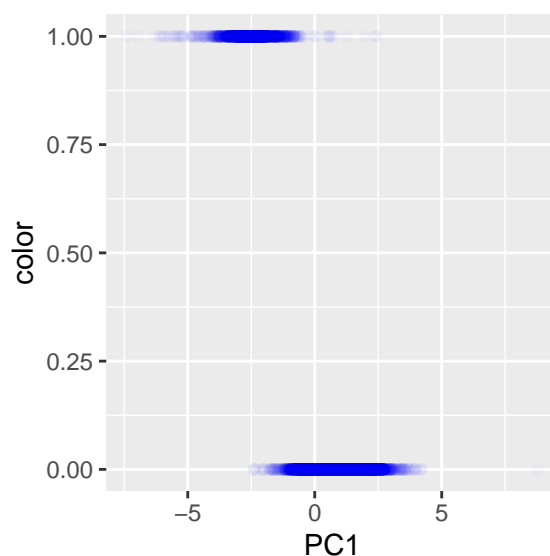
1.2 PCA Model

The summary of pca is:

```
## Importance of components:
##          PC1      PC2      PC3      PC4      PC5      PC6      PC7
## Standard deviation  1.7407 1.5792 1.2475 0.98517 0.84845 0.77930 0.72330
## Proportion of Variance 0.2754 0.2267 0.1415 0.08823 0.06544 0.05521 0.04756
## Cumulative Proportion 0.2754 0.5021 0.6436 0.73187 0.79732 0.85253 0.90009
##          PC8      PC9      PC10     PC11
## Standard deviation  0.70817 0.58054 0.4772 0.18119
## Proportion of Variance 0.04559 0.03064 0.0207 0.00298
## Cumulative Proportion 0.94568 0.97632 0.9970 1.00000
```

(1) PC_1 predicts color

The figure of color V.S PC_1 is listed below. $color == 1$ means red wine; $color == 0$ means white wine. We can see PC_1 has predictive power for $color$ in this figure. PC_1 is easily capable of distinguishing the reds from the whites.



Details of pca model for color(1 component):

```
##
## Call:
## glm(formula = color ~ PC1, family = binomial(), data = pca_data)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -3.1313  -0.0749  -0.0120  -0.0001   5.3871
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -4.7367     0.1841  -25.73  <2e-16 ***
## PC1          -3.9845     0.1517  -26.26  <2e-16 ***
```

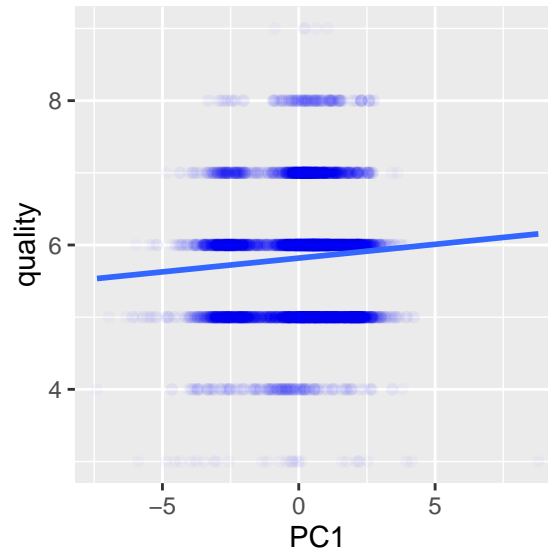
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 7250.98  on 6496  degrees of freedom
## Residual deviance:  935.06  on 6495  degrees of freedom
## AIC: 939.06
##
## Number of Fisher Scoring iterations: 9
```

The f1-score is:

```
## [1] 0.9538267
```

(2) PC_1 predicts quality

The figure of *quality* V.S PC_1 is listed below: PC_1 is not easily capable of predicting quality



Details of pca model for quality(Only one component):

```
##
## Call:
## lm(formula = quality ~ PC1, data = pca_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.1540 -0.7961  0.1421  0.2767  3.2155
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.818378   0.010803  538.575  < 2e-16 ***
## PC1          0.038202   0.006207   6.155  7.97e-10 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8708 on 6495 degrees of freedom
## Multiple R-squared:  0.005798,    Adjusted R-squared:  0.005645
## F-statistic: 37.88 on 1 and 6495 DF,  p-value: 7.972e-10
```

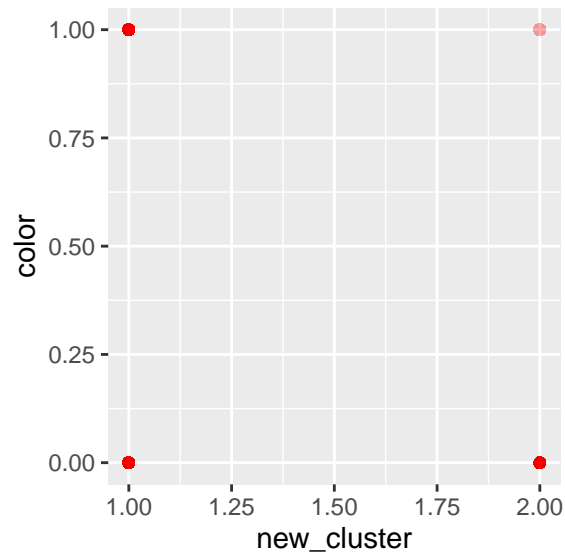
The rmse is:

```
## [1] 0.8706529
```


1.3 Kmeans Model

(1) *Kmeans* predicts color

The figure of color V.S *new_cluster* is listed below. *color* == 1 means red wine; *color* == 0 means white wine. *new_cluster* is not easily capable of distinguishing the reds from the whites.



Details of kmeans model for color(2 classes):

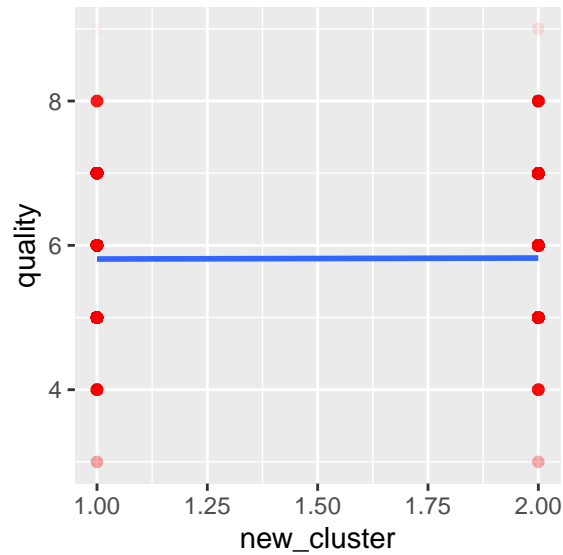
```
##
## Call:
## glm(formula = color ~ new_cluster, family = binomial(), data = wine)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2448  -0.2159  -0.2159  -0.2159   2.7461
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    4.0612     0.1333   30.46  <2e-16 ***
## new_cluster   -3.9042     0.1161  -33.63  <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: 7251.0  on 6496  degrees of freedom
## Residual deviance: 4684.5  on 6495  degrees of freedom
## AIC: 4688.5
##
## Number of Fisher Scoring iterations: 6
```

The f1-score is :

```
## [1] 0.6870887
```

(2) *Kmeans* predicts quality

The figure of *quality* V.S *new_cluster* is listed below.*new_cluster* is not easily capable of predicting quality.



Details of kmeans model for quality:

```
##
## Call:
## lm(formula = quality ~ new_cluster, data = wine)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.8243 -0.8105  0.1757  0.1895  3.1895
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.79674    0.03596 161.199  <2e-16 ***
## new_cluster  0.01380    0.02187   0.631   0.528
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8733 on 6495 degrees of freedom
## Multiple R-squared:  6.131e-05, Adjusted R-squared: -9.265e-05
## F-statistic: 0.3982 on 1 and 6495 DF, p-value: 0.528
```

The rmse is :

```
## [1] 0.8731613
```

1.4 Results and Conclusions

The results of different models are listed in table 1.

Table 1: Comparison between Models

	f1-score(color classification)	rmse(quality prediction)
Benchmark Model	0.9896714	0.7346533
PCA Model(One component)	0.9538267	0.8706529
Kmeans Model(2 classes)	0.6870887	0.8731613

Our target is to see whether the differences in the labels (red/white and quality score) emerge **naturally** from applying an unsupervised technique to the chemical properties. Therefore, we use One component of PCA and 2 classes of Kmeans Model(Reduce the dimension as much as possible), to reflect **naturally**.

From table 1, we can see PCA Model(One component) and Kmeans Model(2 classes) performs equally well in quality prediction, and they are both worse than Benchmark Model.

As for color classification, PCA Model(One component)(0.9538267) is much better than Kmeans Model(2 classes)(0.6870887), as Kmeans Model(2 classes) deletes too much information.

PCA technique makes more sense to me for this data!

PCA technique is easily capable of distinguishing red from white; PCA technique is NOT easily capable of predicting quality. Kmeans technique is Not easily capable of distinguishing red from white; Kmeans technique is NOT easily capable of predicting quality.

2 Market segmentation

We define a group of correlated interests as market segment.

There are 4 unwanted categories “chatter”, “uncategorized”, “adult”, “spam”, and we delete them first. 32 features are left after deletion. Then we calculate the correlation matrix (32×32) of these features.

We define $\text{correlation} \geq 0.5 \Rightarrow 2 \text{ features}(f_1, f_2) \text{ are similar} \Rightarrow f_1 \sim f_2$. We also define if $f_1 \sim f_2, f_2 \sim f_3$, then $f_1 \sim f_3$. We connect them together and get 5 clusters. Features out of these 5 clusters are deleted for simplicity.

Segmentations and advice are listed in table 2.

Table 2: Segmentations and Advice

Segmentations	Description	Advice
"travel", "politics", "computers", "news", "automotive"	middle-aged man who like traveling	our drink relax yourselves
"sports_fandom", "food", "religion", "parenting", "school"	Young parents who raise their kids	our drink is good for kids
"health_nutrition", "outdoors", "personal_fitness"	Old people who care about their health	our drink is good for health
"college_uni", "online_gaming", "sports_playing"	Young students worry about their study.	our drink gives you energy
"cooking", "beauty", "fashion"	Fashionable woman	our drink makes you more beautiful

Mean-values of number of posts of different groups are listed below. People who post more frequently than mean-value of certain group can be classified as that group.

[1] "group 1"

```
##      travel  politics  computers      news  automotive
##  1.5850038  1.7886323  0.6490738  1.2055316  0.8298655
```

[1] "group 2"

```
## sports_fandom      food      religion      parenting      school
##      1.5940117      1.3974879      1.0954073      0.9213398      0.7676986
```

[1] "group 3"

```
## health_nutrition      outdoors  personal_fitness
##      2.5672418      0.7826694      1.4620655
```

[1] "group 4"

```
##      college_uni  online_gaming  sports_playing
##      1.5494798      1.2088302      0.6391779
```

[1] "group 5"

```
##      cooking      beauty      fashion
##  1.9982238  0.7051510  0.9965745
```

3 Association rules for grocery purchases

Pick your own thresholds for lift and confidence

$$lift_{threshold} = 2, confidence_{threshold} = 0.3$$

just be clear what these thresholds are and how you picked them.

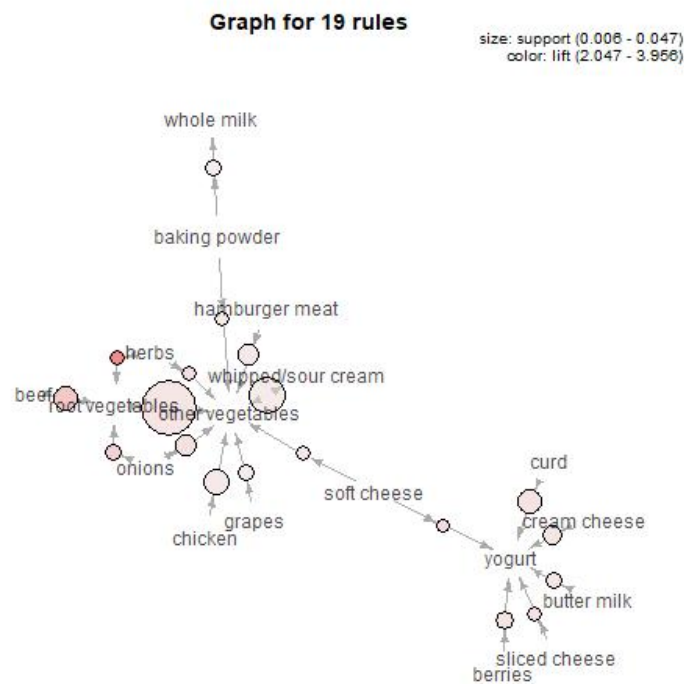
I pick them in order to leave about 100 rules

$$lift(X \rightarrow Y) = \frac{P(Y|X)}{P(Y)} > 2 \quad Confidence(X \rightarrow Y) = \frac{P(XY)}{P(X)} > 0.30$$

Do your discovered item sets make sense? Present your discoveries in an interesting and concise way.

I believe item sets make sense, just look at the following figure.

- (1) Those milk products point to “yogurt.”
- (2) A lot of foods like meat, fruits and vegetables point to “other vegetables”.
- (3) The rules in the following figure are totally consisted of foods.



4 Author attribution

Text data requires lots of preprocessing. This section is divided into 2 parts. Part 1: Preprocess data; Part 2: Model evaluation.

Note: In order to save time, the code of this part has been annotated.

4.1 Part 1: Preprocess data

Train data: 50 authors and 50 essays for each author.

Test data: 50 authors and 50 essays for each author.

A part of Author names are as follows:

AaronPressman	2021/5/5 20:21	文件夹
AlanCrosby	2021/5/5 20:21	文件夹
AlexanderSmith	2021/5/5 20:21	文件夹
BenjaminKangLim	2021/5/5 20:21	文件夹
BernardHickey	2021/5/5 20:21	文件夹
BradDorfman	2021/5/5 20:21	文件夹
DarrenSchuettler	2021/5/5 20:21	文件夹
DavidLawder	2021/5/5 20:21	文件夹
EdnaFernandes	2021/5/5 20:21	文件夹
EricAuchard	2021/5/5 20:21	文件夹
FumikoFujisaki	2021/5/5 20:21	文件夹
GrahamEarnshaw	2021/5/5 20:21	文件夹
HeatherScofield	2021/5/5 20:21	文件夹
JaneMacartney	2021/5/5 20:21	文件夹
JanLopatka	2021/5/5 20:21	文件夹
JimGilchrist	2021/5/5 20:21	文件夹

A part of essay is as follows:

nvestors smiled on the bourses of central and eastern Europe as brightly as the summer sun this week, though there were some indications these may by little more than fair-weather friends to the markets. Exchanges in Prague, Warsaw, Budapest, Bratislava and Bucharest all gained ground, while Zagreb and Sofia traded mixed. Ljubljana was the one gray cloud, posting slight losses. The Central European Share Index (CESI) which reflects the price movements of 50 selected Czech, Polish and Hungarian shares, firmed 66.11 points. PRAGUE

Data preprocess is as follows:

- (1) We load the train data as a data frame which associates authors and essays together.

	text	author
1	The Internet may be overflowing with new technology but c...	AaronPressman
2	The U.S. Postal Service announced Wednesday a plan to bo...	AaronPressman
3	Elementary school students with access to the Internet learn...	AaronPressman
4	An influential Internet organisation has backed away from a ...	AaronPressman
5	An influential Internet organisation has backed away from a ...	AaronPressman
6	A group of leading trademark specialists plans to release re...	AaronPressman
7	When a company in California sells a book to a consumer in...	AaronPressman
8	U.S. laws governing the trillion dollar futures markets could ...	AaronPressman
9	Supreme Court justices Wednesday sharply questioned rule...	AaronPressman

- (2) We remove Punctuation, stopwords, numbers and Whitespace. In addition, all letters are changed to lower case.
- (3) Some words in test data set may not appear in train data. In order to address this problem, we select common_words.

$$common_words = words_of_train_data \cap words_of_test_data$$

- (4) Finally, one-hot coding is adopted. It converts numerical value to “True” and “False”. Just like the following figures.

	access	accounts	advocacy	agencies	alliance	also	announced	artists
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	1	0	0
3	3	0	0	0	0	2	1	0
4	0	0	0	0	0	0	0	0

	access	accounts	advocacy	agencies	alliance	also	announced	artists
1	False	False	False	False	False	False	False	False
2	False	False	False	False	False	True	False	False
3	True	False	False	False	False	True	True	False
4	False	False	False	False	False	False	False	False

4.2 Part 2:Model evaluation

Many models have been tried. However, due to the size of our data, only Naive Bayesian Model can effectively fit on the data and make prediction.

The accuracy on test data is 67.4%.