

Group Project

Toshiki Arita
Arthur Pang
Emily Sison
Jennifer Chen

1 Problem 1

1.a Part A - compare two means

In this problem we will be comparing the mean of registered bike users on non-holidays and holidays.

H1 will be the alternative hypothesis and H0 the null hypothesis

H1 : The mean of registered bike users on non holidays and holidays are not equal.

H0 : The mean of registered bike users on non holidays and holidays are equal.

If we define the mean of registered bike users on a non-holiday as u_1 and holidays as u_2 then our equations would be.

$$\begin{aligned} H1 : u_1 &\neq u_2 \\ H0 : u_1 &= u_2 \end{aligned} \tag{1}$$

After running a t-Test we have the following results

$$\begin{aligned} \text{mean of registered bike users on} \\ \text{a non holiday} &= 3685.332 \\ \text{holiday} &= 2670.286 \end{aligned} \tag{2}$$

$$\text{Difference} = 3685.332 - 2670.286 = 1015.046 \tag{3}$$

The difference tells us that on average for this sample, there were more registered bike users on non holidays.

The 95% confidence interval is 327.3991 to 1702.6943. This interval tells us something about the difference between means of registered bikers on holidays vs non holidays for the population (instead of sample). We can say there is a 95% chance that the confidence interval calculated contains the true mean of the population.

$$327.3991 \leq \text{true difference mean} \leq 1702.6943 \tag{4}$$

This would satisfy our hypothesis H1 where the population difference is bigger than zero.

So we can say that based on this sample, there is a difference between the number of registered bike users on a holiday vs non holidays

One thing to note however though is that we can accept H_1 based on our sample because the confidence interval doesn't include 0. If the interval includes 0 it's possible for the null hypothesis, H_0 , to be true but at the same time allows H_1 to be true as well. This leads to a result that doesn't really tell us anything about the population since we can't accept or reject either hypothesis.

1.b Part B - compare two proportions

In this section we'll compare the proportions of weather situation 1 on working days vs non working days.

Working days :

$$\begin{aligned} n1 &= 500 \text{ working days} \\ y1 &= 307 \text{ days had weather 1} \\ \hat{p}1 &= (307/500) = 0.614 \end{aligned} \tag{5}$$

Non working days :

$$\begin{aligned} n2 &= 231 \text{ non working days} \\ y2 &= 156 \text{ days had weather 1} \\ \hat{p}2 &= (156/231) = 0.675 \end{aligned} \tag{6}$$

Is there a significant difference between weather situations on working vs nonworking days?

$$\begin{aligned} H1 : p1 &\neq p2 \text{ (There is a difference)} \\ H0 : p1 &= p2 \text{ (There is no difference)} \end{aligned} \tag{7}$$

$$\text{Difference} = \hat{p}2 - \hat{p}1 = 0.061 \tag{8}$$

The overall sample proportion of weather situation 1 days can be defined as \hat{p}

$$\begin{aligned} \hat{p} &= (307 + 156)/731 \\ &= 0.6337 \end{aligned} \tag{9}$$

The standard error in this sample was/

$$s.e = \sqrt{\hat{p}1(1 - \hat{p}2)\left(\frac{1}{n1} + \frac{1}{n2}\right)} = 0.03834 \tag{10}$$

Now we can find our Z-value

$$\begin{aligned} Z &= \frac{\hat{p}2 - \hat{p}1}{s.e} \\ &= 1.59 \end{aligned} \tag{11}$$

If we look up $z = 1.59$ on the Z table, we can see that the p value is $1 - 0.9441 = 0.0559$. The p value is larger than 0.05 and we can also see that $-1.96 \leq z \leq 1.96$ allowing us to accept the null hypothesis, that there is no difference between the weather situation on working days vs non working days, for the population.

2 Problem 2

2.a

Our response variable is the number of people riding a bike and our predictor variable is the wind speed. Our first model predicts the effects of wind speed alone on the mean number of total people riding a bike. We divide the data into a training set the size of the original data and a validation set of the original data.

2.b

The training set predicts the response variable from the predictor variables.

```
1 > nr = nrow(day)
2 > train = sample(1:nr, floor(2/3 * nr), replace = FALSE)
3 > val = setdiff(1:nr, train)
4 > lmout = lm(day[train, 16] ~ day[train, 13])
5 > summary(lmout)
6
7      Call:
8      lm(formula = day[train, 16] ~ day[train, 13])
9
10     Residuals:
11             Min       1Q   Median       3Q      Max
12    -4222.0  -1359.6   -65.9   1292.0   4338.4
13
14     Coefficients:
15                                     Estimate Std. Error t value Pr(>|t|)
16 (Intercept)             5617.2         213.3   26.339 < 2e-16 ***
17 day[train, 13]        -6153.8         1029.2   -5.979 4.35e-09 ***
18
19     Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
20
21     Residual standard error: 1819 on 485 degrees of freedom
22     Multiple R-squared:  0.06866,    Adjusted R-squared:  0.06674
23     F-statistic: 35.75 on 1 and 485 DF,  p-value: 4.348e-09
```

2.c

In the linear parametric model, $mW;H(t) = ct + d$, c is our estimate coefficient in the highwind row, t is a normalized wind speed, and d is the y-intercept of the slope. c and d are population parameters.

In our linear model function calls the wind speed data from column 13 from our data and the count or number of people riding a bike from column 16. The estimate coefficient of highwind is the slope of the wind speed. It is a negative slope, so as wind speed increases, the amount of riders decrease. The standard error coefficient is lower than our estimate coefficient. Since the standard error isn't a magnitude lower than our estimate, we have some variability in the estimate coefficient.

The validation set shows how well our model predicts new data.

```
1 > val = setdiff(1:nr, train)
2 > val
3
4      1]  1  3  4  8  9 11 13 18 20 22 29 30 31 32 33 34 35 36 37 38
      39 40 41 42 43 44 45 46 47 48 49 50 51 52
```

2.d

We call our linear model function, but adding the temperature predictor and adding the quadratic t^2 (where we square the wind speed), and the

```
1 > temp = day
2 > highwind = day
3 > nr = nrow(day)
4 > train = sample(1:nr, floor(2/3 * nr), replace = FALSE)
5 > val = setdiff(1:nr, train)
6 > lmout = lm(highwind[train, 16] ~ highwind[train, 13] + temp[train, 10]
+ highwind[train, 13]*temp[train, 10] + I($highwind[train, 13]^2$))
7 > summary(lmout)
8
9      Call:
10      lm(formula = highwind[train, 16] ~ highwind[train, 13] + temp[
train,
11      10] + highwind[train, 13] * temp[train, 10] + I($
highwind[train,
12      13]^2$))
13
14      Residuals:
15              Min          1Q      Median          3Q             Max
16      -3672.7  -1133.5   -187.7   1151.2   3894.6
17
18      Coefficients:
19
```

Est

```

20      (Intercept)                2798.5        716.9        3.903
      0.000108 ***
21      highwind[train , 13]        -10162.9       5054.5       -2.011
      0.044917 *
22      temp[train , 10]            5529.6        1138.6        4.856
      1.62e-06 ***
23      I($highwind[train , 13]^2$)      8917.5        8488.9
      1.050 0.294016
24      highwind[train , 13]:temp[train , 10]  5776.7        5806.8        0.995
      0.320321
25
26      Signif. codes:  0      ***      0.001      **      0.01      *      0.05
      .      0.1      1
27
28      Residual standard error: 1525 on 482 degrees of freedom
29      Multiple R-squared:  0.4099,    Adjusted R-squared:  0.405
30      F-statistic: 83.71 on 4 and 482 DF,  p-value: < 2.2e-16

```

The estimate coefficient of highwind is the slope of the wind speed. The estimate coefficient of $I(\text{highwind}[\text{train}, 13]^2)$ is the slope of the wind speed squared.

The estimate coefficient $\text{highwind}[\text{train}, 13]:\text{temp}[\text{train}, 10]$ is the slope of the highwind predictor multiplied by the temperature predictor. All coefficients except for highwind have a positive slope. The higher the wind speed, the less people ride their bikes. The higher the temperature, the more people ride their bikes. When the wind and temperature increases, the more people ride their bikes.

The standard error coefficient for temperature is lower than our estimate coefficient. Since the standard error isn't larger than our estimate, we have less variability in the estimate coefficient. The standard error coefficients for wind speed, wind speed squared, and temperature multiplied by wind is larger than our estimate coefficient. So we have more variability in the estimate coefficient. Since the standard error is large, this model is prone to errors in the three previous areas and thus fairly inaccurate.

2.e

```

1 > temp = day
2 > highwind = day
3 > weathersit = day
4 > nr = nrow(day)
5 > val = setdiff(1:nr, train)
6 > train = sample(1:nr, floor(2/3 * nr), replace = FALSE)
7 > lmout = lm(highwind[train , 16] ~ highwind[train , 13] + temp[train ,
      10] + weathersit[train , 9])
8 > summary(lmout)
9
10      Call:
11      lm(formula = highwind[train , 16] ~ highwind[train , 13] + temp[
      train ,
12      10] + weathersit[train , 9])
13

```

```

14      Residuals:
15          Min       1Q   Median       3Q      Max
16      -3801.9  -1116.9   -130.9   1171.5   3807.0
17
18      Coefficients:
19
20                      Estimate Std. Error t
21                      value Pr(>|t|)
22      (Intercept)          3462.9      328.8  10.532 < 2e-16 ***
23      highwind[train , 13]    -3652.8      879.7   -4.152 3.90e-05 ***
24      temp[train , 10]        5789.1      370.1  15.642 < 2e-16 ***
25      weathersit[train , 9]    -812.6      123.4   -6.587 1.17e-10 ***
26
27      Signif. codes:  0      ***    0.001    **    0.01    *    0.05
28                      .      0.1      1
29
30      Residual standard error: 1463 on 483 degrees of freedom
31      Multiple R-squared:  0.4267,    Adjusted R-squared:  0.4231
32      F-statistic: 119.8 on 3 and 483 DF,  p-value: < 2.2e-16

```

Our final model is composed of the predictors: wind speed, temperature, and weather situation. We chose these predictors because they all relate to the weather and they are all significant. We wanted to see how weather affects the number of people riding their bikes. If we were to test at a significance level of 0.01 are predictors are significant because their p-values translate to a significant level of 0.001 or lower. Since the p-values are small there is a strong evidence against the null hypothesis. The null hypothesis is that the predictors don't affect the mean of people riding their bikes at certain significance level, such as 0.01. They all have a small standard deviation so they are all fairly accurate estimates.

After trying multiple predictor combinations, we found that this model set was the most accurate. For example, in the model below, work day is less significant so it is less correlated to the number of people riding their bikes compared to wind speed or temperature. The p-value of work day translates to a significance level of approximately 0.05. Because we want to test our hypothesis at a significance level at 0.01, we eliminated work day from our final model.

```

1 > workday = day
2 > highwind = day
3 > season = day
4 > nr = nrow(day)
5 > val = setdiff(1:nr, train)
6 > train = sample(1:nr, floor(2/3 * nr), replace = FALSE)
7 > lmout = lm(highwind[train , 16] ~ highwind[train , 13] + workday[train ,
8     8] + season[train , 3])
9 > summary(lmout)
10
11      Call:
12      lm(formula = highwind[train , 16] ~ highwind[train , 13] +
13          workday[train , 8] + season[train , 3])
14
15      Residuals:

```

```

15           Min      1Q  Median      3Q      Max
16    -4933.5 -1355.5    -83.1  1275.8  4450.0
17
18    Coefficients:
19
20                                     Estimate Std. Error t
21                                     value Pr(>|t|)
20    (Intercept)                3181.8      324.4    9.808 < 2e-16 ***
21    highwind[train, 13]      -3005.8      1001.7   -3.001  0.00283 **
22    workday[train, 8]         415.5       172.8    2.404  0.01658 *
23    season[train, 3]         608.7       72.2     8.431 3.98e-16 ***
24    -----
25    Signif. codes:  0    ***    0.001    **    0.01    *    0.05
26                    .    0.1      1
27
28    Residual standard error: 1757 on 483 degrees of freedom
29    Multiple R-squared:  0.1691,    Adjusted R-squared:  0.164
30    F-statistic: 32.77 on 3 and 483 DF,  p-value: < 2.2e-16

```

2.f

```

1 > val = setdiff(1:nr, train)
2 > val
3
4      [1]  2  5  8 10 15 17 19 25 26 29 30 31 33 34 39
5      [31] 41 45 46 54 56 58 60 61 64 68 71 74 76 80 81
6      [61] 88 90 93 98 100 104 110 112 118 122 126 127 129 130
7      [91] 135 136 137 143 151 152 154 155 157 158 161 162 166 172 174
8      [121] 178
9      [151] 180 181 187 188 189 191 193 195 196 198 205 209 210 214
10     [181] 220 223 225 229 230 231 237 238 239 241 244 245 250 253 256
11     [211] 258
12     [241] 259 260 261 282 286 287 291 298 300 302 305 306 310 314
13     [271] 316 317 327 328 331 332 338 340 341 343 346 349 350 355 362
14     [301] 364
15     [331] 367 371 380 383 386 390 393 398 399 401 403 406 410 416
16     [361] 418 419 423 424 427 429 431 435 437 438 439 442 446 453 455
17     [391] 456
18     [421] 458 461 462 467 470 476 477 480 485 486 487 489 490 491
19     [451] 493 498 502 506 507 508 516 517 520 522 524 525 529 531 533
20     [481] 544
21     [511] 547 551 553 562 565 566 567 569 571 572 577 582 583 584
22     [541] 587 589 592 595 598 602 603 606 607 609 611 621 624 625 626
23     [571] 628
24     [601] 633 635 643 645 647 648 651 653 654 657 661 664 665 667
25     [631] 676 678 679 686 690 694 699 700 701 704 705 707 709 710 711
26     [661] 712
27     [691] 713 714 719 726

```

The adjusted r^2 value in our final model is 0.4231. When r^2 is closer to 1, then the more mean values fall on our best-fit line. If it is closer to 0, then our mean values are more spread out from

our best-fit line. The best-fit line represents the trend of the predictions.

In the below function, we estimate our linear model and use that data to find the estimate of the mean. sum is

```

1 temp = day
2 highwind = day
3 weathersit = day
4 nr = nrow(day)
5 val = setdiff(1:nr, train)
6 val
7
8 train = sample(1:nr, floor(2/3 * nr), replace = FALSE)
9 lmout = lm(highwind[train, 16] ~ highwind[train, 13] + temp[train, 10]
10 + weathersit[train, 9])
11 summary(lmout)
12 lmout
13 newy = predict(lmout, day[val, c(13,10,9)]) *not working*
14 newxl = day[val[1], c(13,10,9)]
15 newxl = as.numeric(newxl)
16 newxl = c(1,newxl)
17 betas = lmout$coef
18 betas %*%newxl
19 newy[1]
20
21 estimate<-function(wind,temp,weath){
22   meanhat<-3462.9+(-3652.8*wind)+(5789.1*temp)+(-812.6*weath)
23   return(meanhat)
24 }
25 bike = day
26 sum<-0
27 sum1<-0
28 for(i in 1:731){
29   if ( sum(val == i)){
30     wnd<-bike[i,13]
31     temp<-bike[i,10]
32     weath<-bike[i,9]
33     sum<-sum+estimate(wnd,temp,weath)
34     sum1<-sum1+bike[i,16]
35   }
36 }
37 sum/244
38 [1] 4431.787
39 sum1/244
40 [1] 4477.906

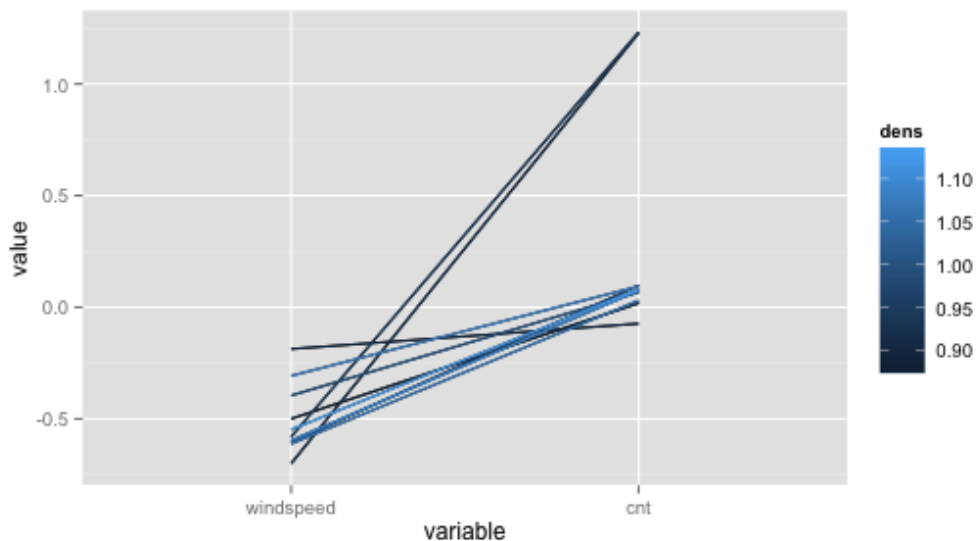
```

sum is the estimated mean sum1 is the actual mean. Based on the regression function we obtained from the training set, we see that the estimated mean of the validation set is close to the actual mean by about 1

2.g

Plot the 10 highest-density values with k=10, windspeed (row 13) vs cnt (row 16)

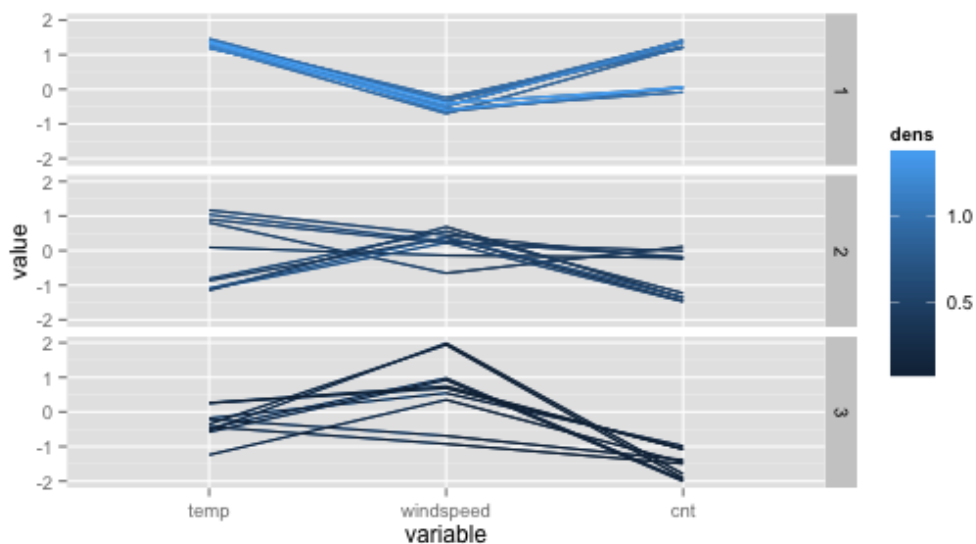
```
1 library(freqparcoord)
2 freqparcoord(day,10,c(13,16),k=10)
```



There seems to be some trend, but its not clear with so few variables.

Next, group the data by weather and plot temp, windspeed, and cnt Weather: 1 - clear, few clouds
2 - mist, cloudy 3 - light snow, light rain

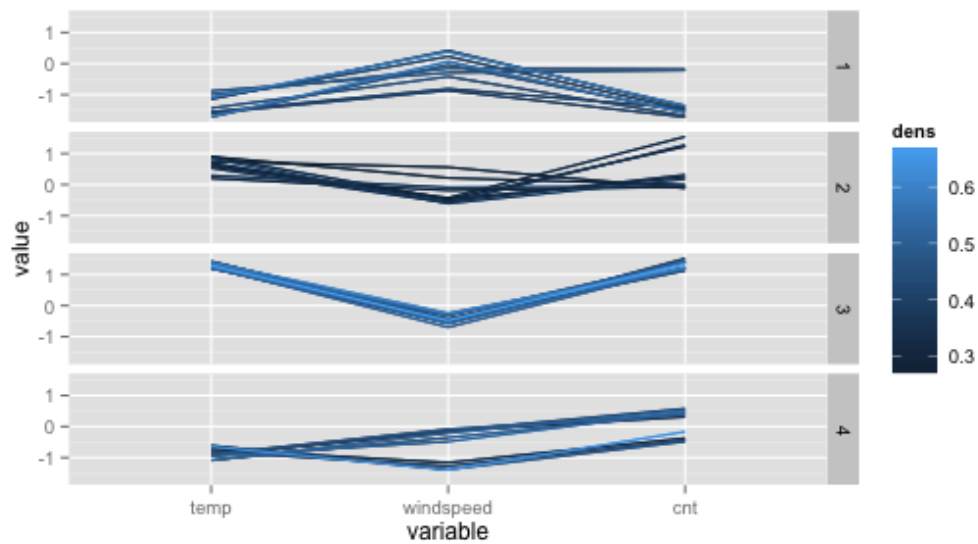
```
1 freqparcoord(day,10,c(10,13,16),9,k=10,method="maxdens")
```



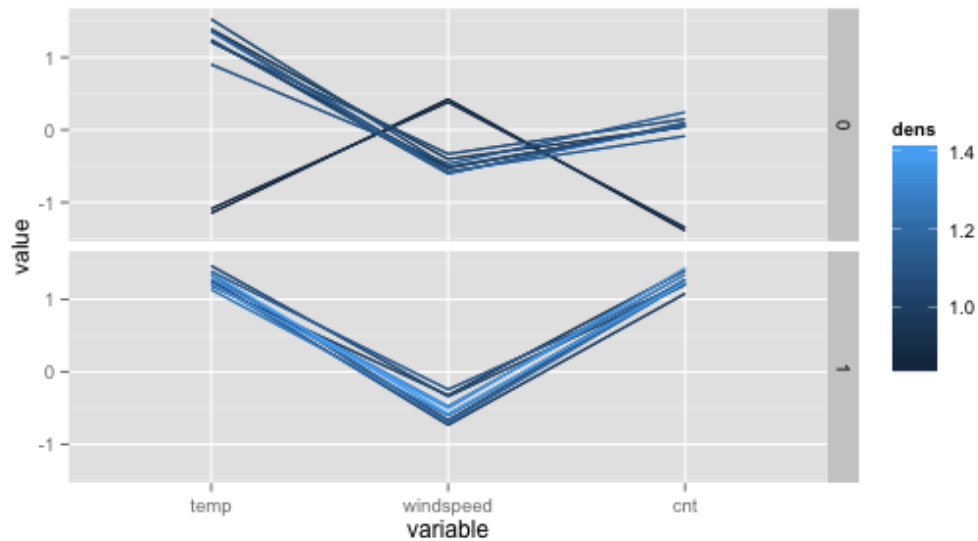
We see a pattern emerging. More people seem to bike on days with good weather than bad weather.

Next, group by season. 1 - spring 2 - summer 3 - fall 4 - winter

```
1 freqparcoord(day,10,c(10,13,16),3,k=10,method="maxdens")
```



There seems to be two different groups, especially when looking at the pattern in fall. After trying out several different variables, we found that the year was a determining factor.

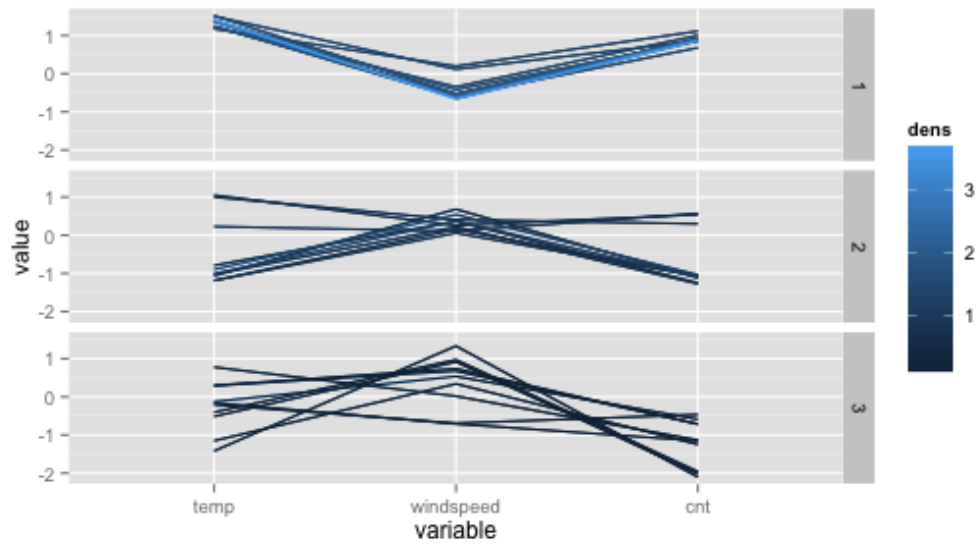


There seems to be distinct patterns between the two years. Therefore it seems appropriate to analyzing each year by itself.

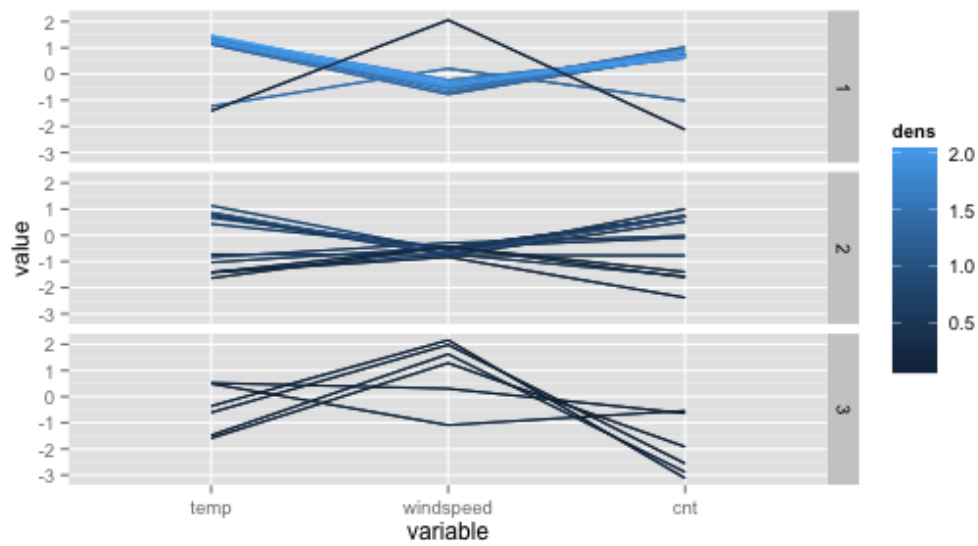
```
1 firstyear = day[day$yr == 0,]
2 secondyear = day[day$yr == 1,]
```

Again grouping by weather,

```
1 freqparcoord(firstyear,10,c(10,13,16),9,k=5,method="maxdens")
```

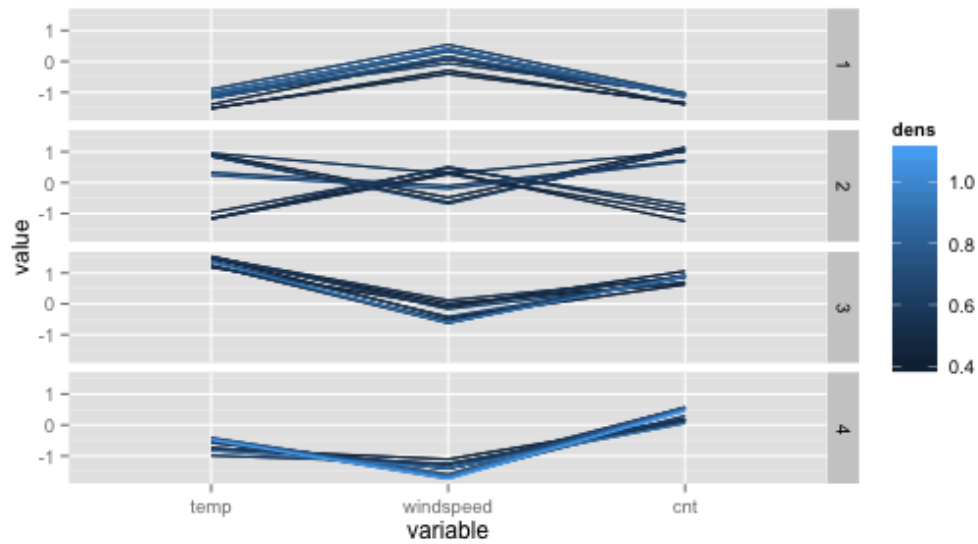


```
1 freqparcoord(secondyear,10,c(10,13,16),9,k=5,method="maxdens")
```

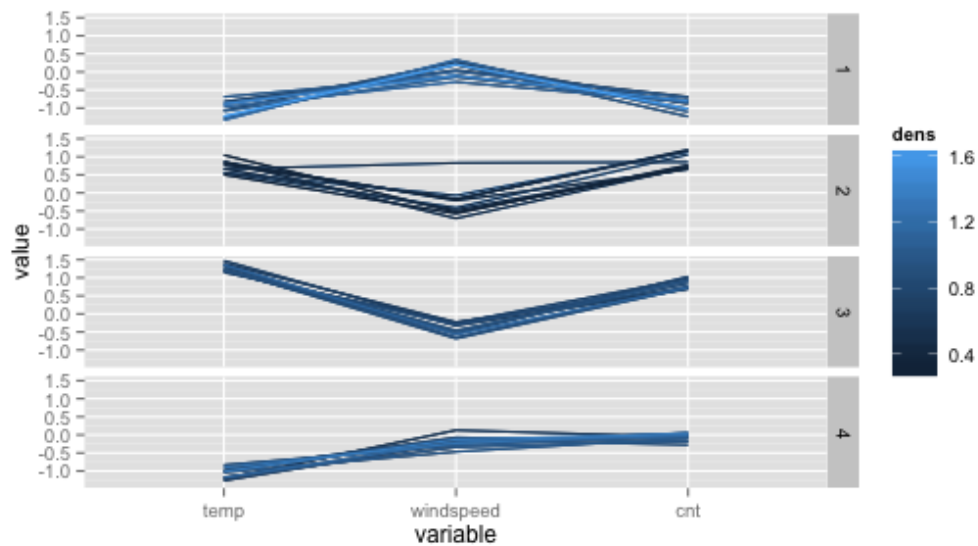


Grouping by season:

```
1 freqparcoord(firstyear,10,c(10,13,16),3,k=5,method="maxdens")
```



```
1 freqparcoord(secondyear,10,c(10,13,16),3,k=5,method="maxdens")
```



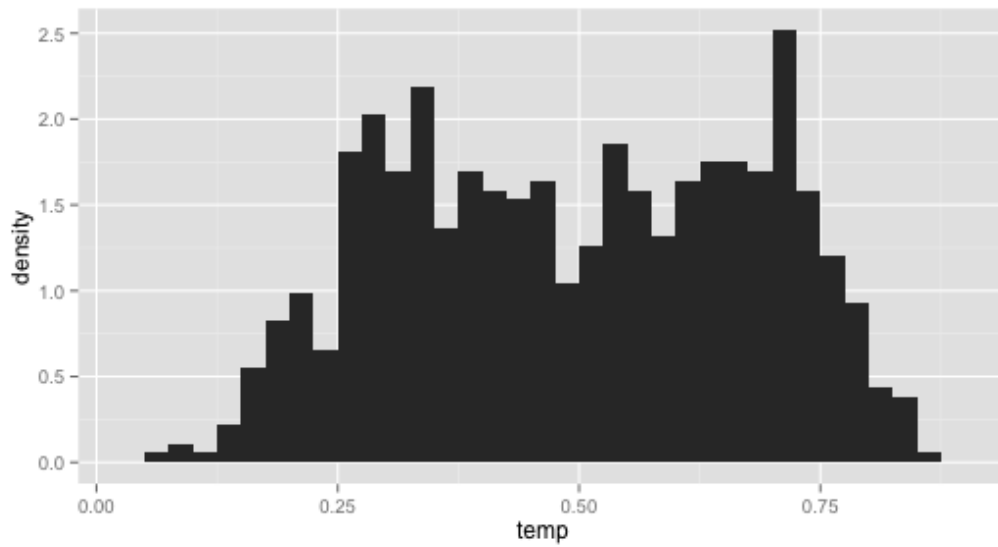
In general, it seems that there is a strong correlation between temperature and windspeed, when grouped by seasons. High temp usually means low windspeed and more bikers.

3 Problem 3

3.a

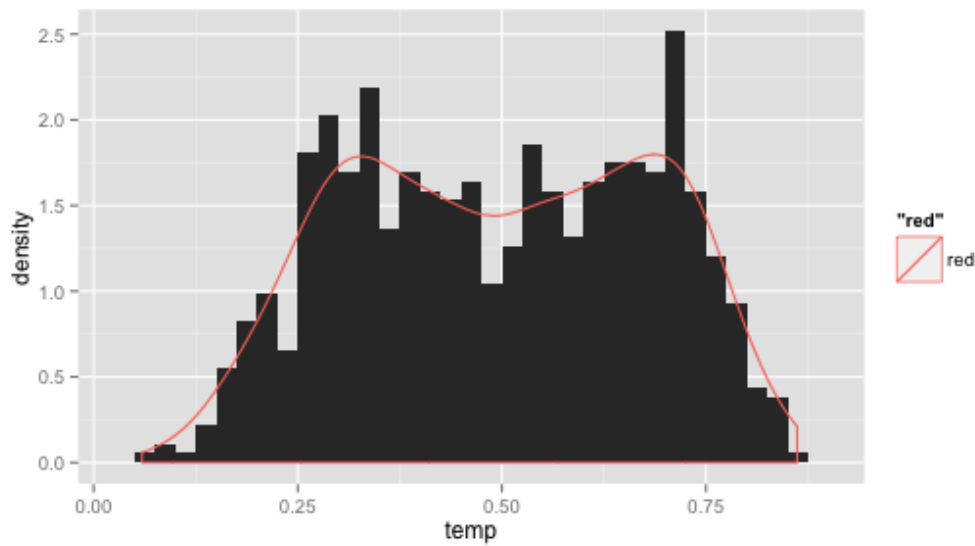
A histogram is a nonparametric estimate of the probability density function. We generate one of temperature using `geom_histogram()` from `ggplot2`:

```
1 day <- read.csv("~/Desktop/132/Final/day.csv")
2 library(ggplot2)
3 ggplot(day) + geom_histogram(aes(x=temp, y=..density..), binwidth=.025)
```



We then used `geom_density()` from `ggplot2` for an estimate of the density function (`geom_density()` uses `stat_density()`, which is a 1d kernel density estimate).

```
1 ggplot(day) + geom_histogram(aes(x=temp, y=..density..), binwidth=.025)
  + geom_density(aes(x=temp, colour="red"))
```



3.b

For the Method of Moments, the parameters are

$$EX = \mu$$

$$E(X^2) = \mu^2 + \sigma^2$$

The sample moments are

$$\frac{1}{N} \sum X_i = \hat{\mu}$$

$$\frac{1}{N} \sum X_i^2 = \hat{\mu} + \hat{\sigma}^2$$

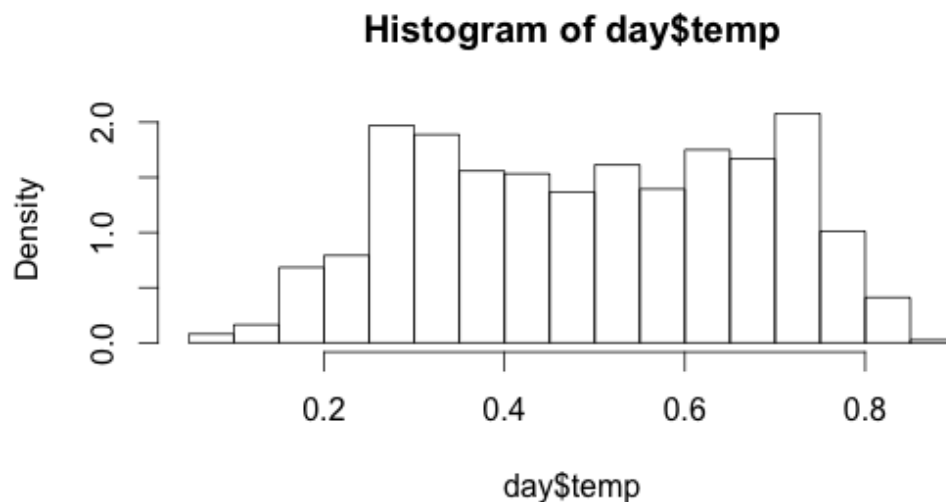
3.c

The graphs in Part A seem to indicate that there is a bimodal distribution. We use the `mixtools` package to fit the data with a mixture model of two Gaussian distributions.

```
1 library(mixtools)
2 mixmdl = normalmixEM(day$temp)
3 str(mixmdl)
4 List of 9
5 $ x      : num [1:731] 0.344 0.363 0.196 0.2 0.227 ...
6 $ lambda : num [1:2] 0.554 0.446
7 $ mu     : num [1:2] 0.359 0.666
8 $ sigma  : num [1:2] 0.1114 0.0858
9 $ loglik : num 265
10 $ posterior : num [1:731, 1:2] 0.999 0.998 1 1 1 ...
11 ..- attr(*, "dimnames")=List of 2
12 .. ..$ : NULL
13 .. ..$ : chr [1:2] "comp.1" "comp.2"
14 $ all.loglik: num [1:182] -167 199 212 219 227 ...
15 $ restarts : num 0
16 $ ft       : chr "normalmixEM"
17 - attr(*, "class")= chr "mixEM"
```

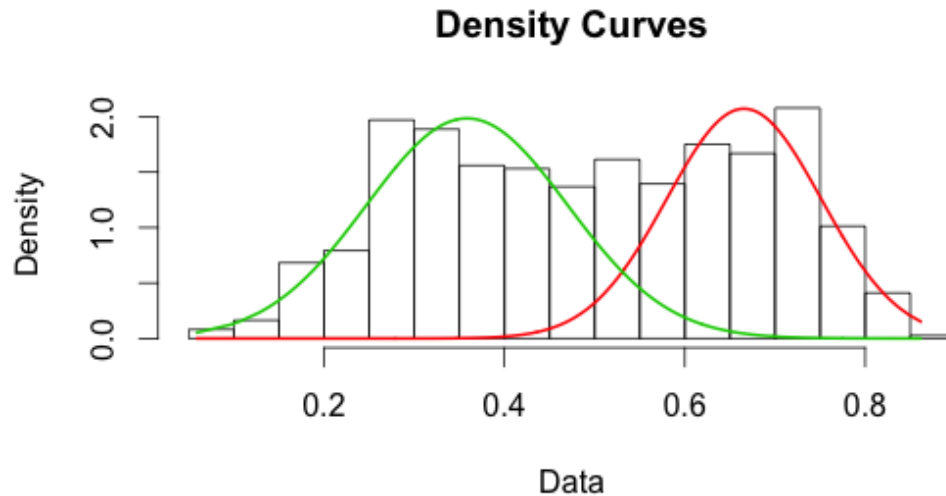
Mixtools uses the EM algorithm. From the data we see that the algorithm estimates the first Gaussian distribution (drawn in red) to have a mean of 0.359 and a standard deviation of 0.1114, and that about 55.4% of the temperatures are of this type. The other 44.6% (drawn in green) have a mean of 0.666 and a standard deviation of 0.0858.

The histogram is again plotted below:



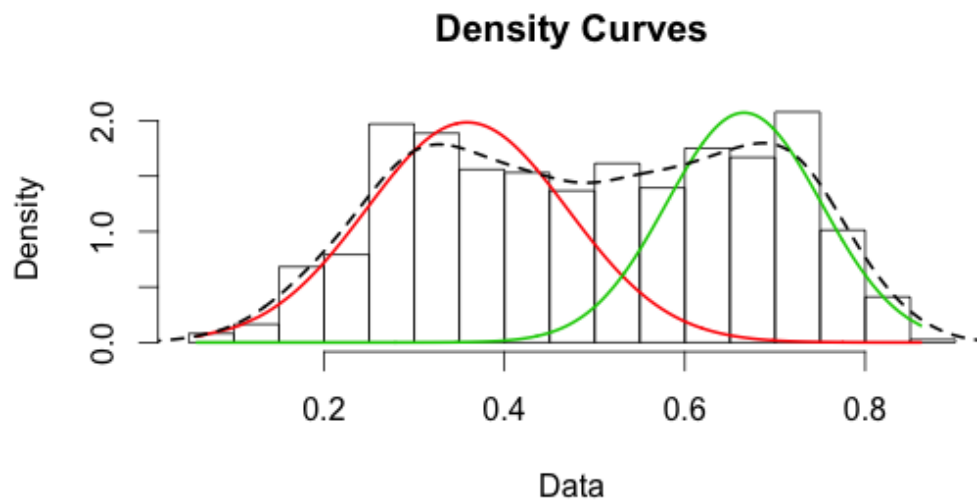
Superimposing the two normal distributions:

```
1 plot(mixmdl, breaks=25, which=2)
```



Adding the density estimation:

```
1 lines(density(day$temp), lty=2, lwd=2)
```



This result seems to make sense, since the temperature is taken over a two-year period, and the mean temperature during summer will be higher than the mean temperature during winter. This analysis suggests that over the course of a year, the density function of temperature follows a mixture of two Gaussian distributions.