# Assignment #1 – Machine Learning – Professor Haugh

## **Jaime Gacitua**

## **jg3499**

## **Monday February 1, 2016**

Problem 1. Read

Problem 2.

1. Looking at the documentation, the variable of interest is **spam**, representing for every line “yes if spam; else no”.

There is a total of 2171 rows in the database, and 21 columns.

As we can see in the summary table below, there are 1.353 values missing in the “spampct” column. There are 13 categorical or binary variables, and there are 8 numerical columns.

isuid id **day.of.week** time.of.day size.kb **box domain** **local**

Min. : 1.000 Min. : 1.0 Fri:316 Min. : 0.00 Min. : 0.00 no :1066 edu :1037 no :1221

1st Qu.: 4.000 1st Qu.: 33.0 Mon:322 1st Qu.: 9.00 1st Qu.: 2.00 yes:1105 com : 807 yes: 950

Median : 9.000 Median : 62.0 Sat:136 Median :12.00 Median : 4.00 net : 118

Mean : 9.234 Mean : 201.2 Sun:178 Mean :12.26 Mean : 16.49 de : 45

3rd Qu.:14.000 3rd Qu.: 108.0 Thu:398 3rd Qu.:16.00 3rd Qu.: 7.00 uk : 42

Max. :19.000 Max. :3470.0 Tue:360 Max. :23.00 Max. :1337.00 org : 26

Wed:461 (Other): 96

digits **name** cappct special **credit sucker porn chain username**

Min. : 0.000 empty : 295 Min. :0.0000 Min. : 0.000 no :2081 no :1945 no :2144 no :2113 no :2080

1st Qu.: 0.000 name :1611 1st Qu.:0.0600 1st Qu.: 0.000 yes: 90 yes: 226 yes: 27 yes: 58 yes: 91

Median : 0.000 single: 265 Median :0.1280 Median : 1.000

Mean : 0.591 Mean :0.1584 Mean : 1.397

3rd Qu.: 0.000 3rd Qu.:0.2000 3rd Qu.: 2.000

Max. :23.000 Max. :1.0000 Max. :35.000

**large.text** spampct **category** **spam**

no :1758 Min. : 0.00 com :635 no :1461

yes: 413 1st Qu.:11.00 list:690 yes: 710

Median :47.50 news:100

Mean :44.63 ord :746

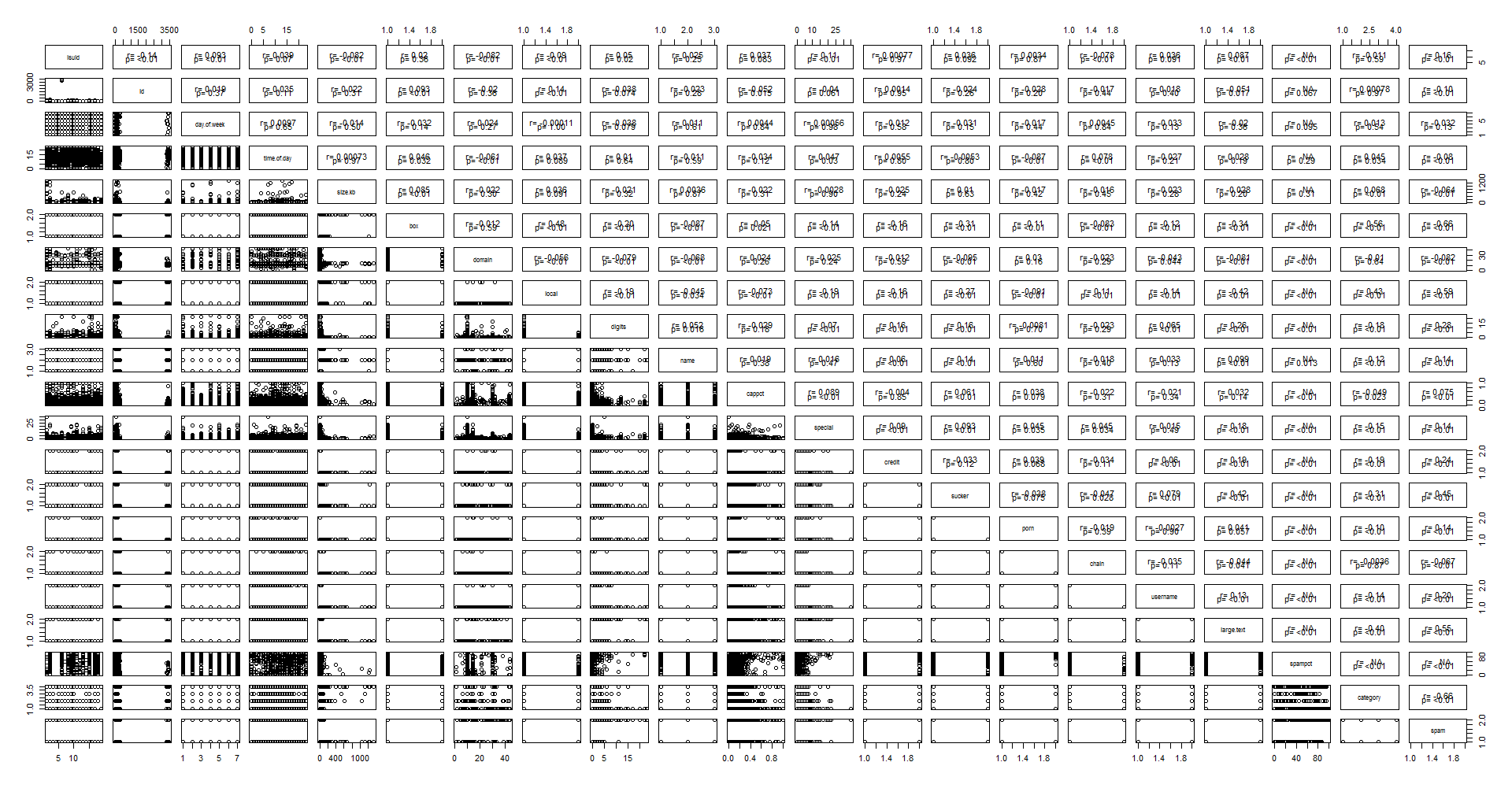
3rd Qu.:76.00

Max. :99.00

**NA's :1353**

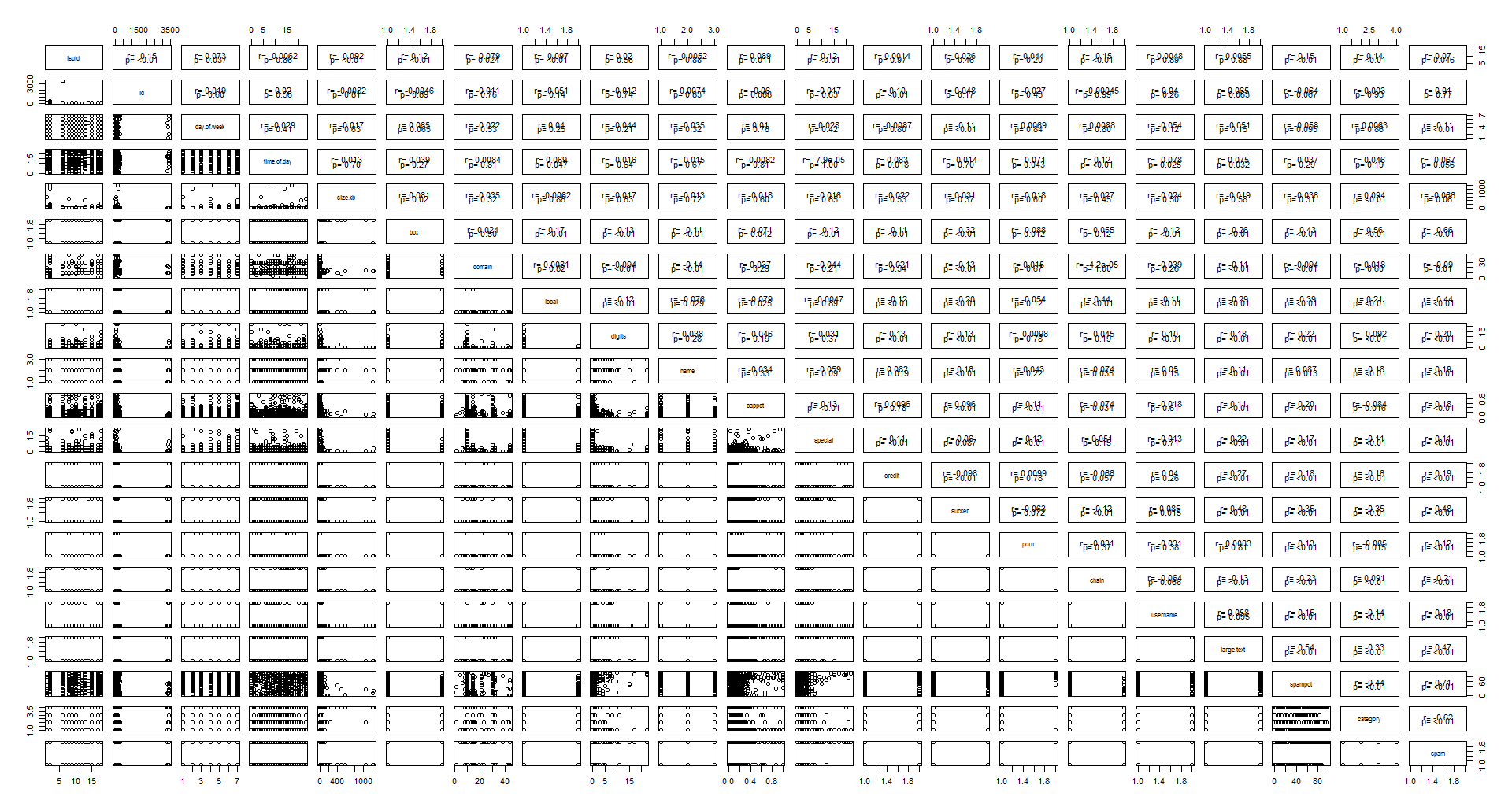
Scatter Plot Matrix for the 2171 rows of the database. Spampct will not display values because it has NA values included.

It results important to notice how the “spam” variable has a correlation of more than 0.45 with the variables, “box”, “local”, “sucker”, “large.text” and “category”.

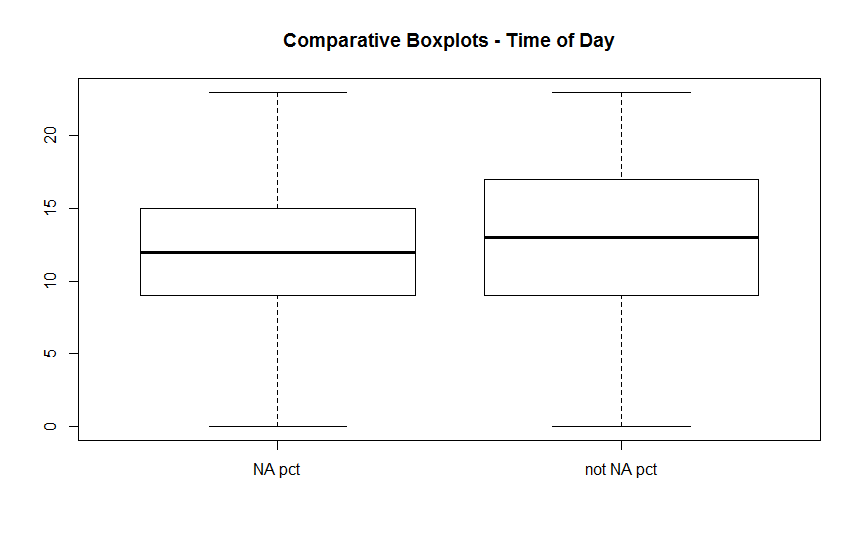


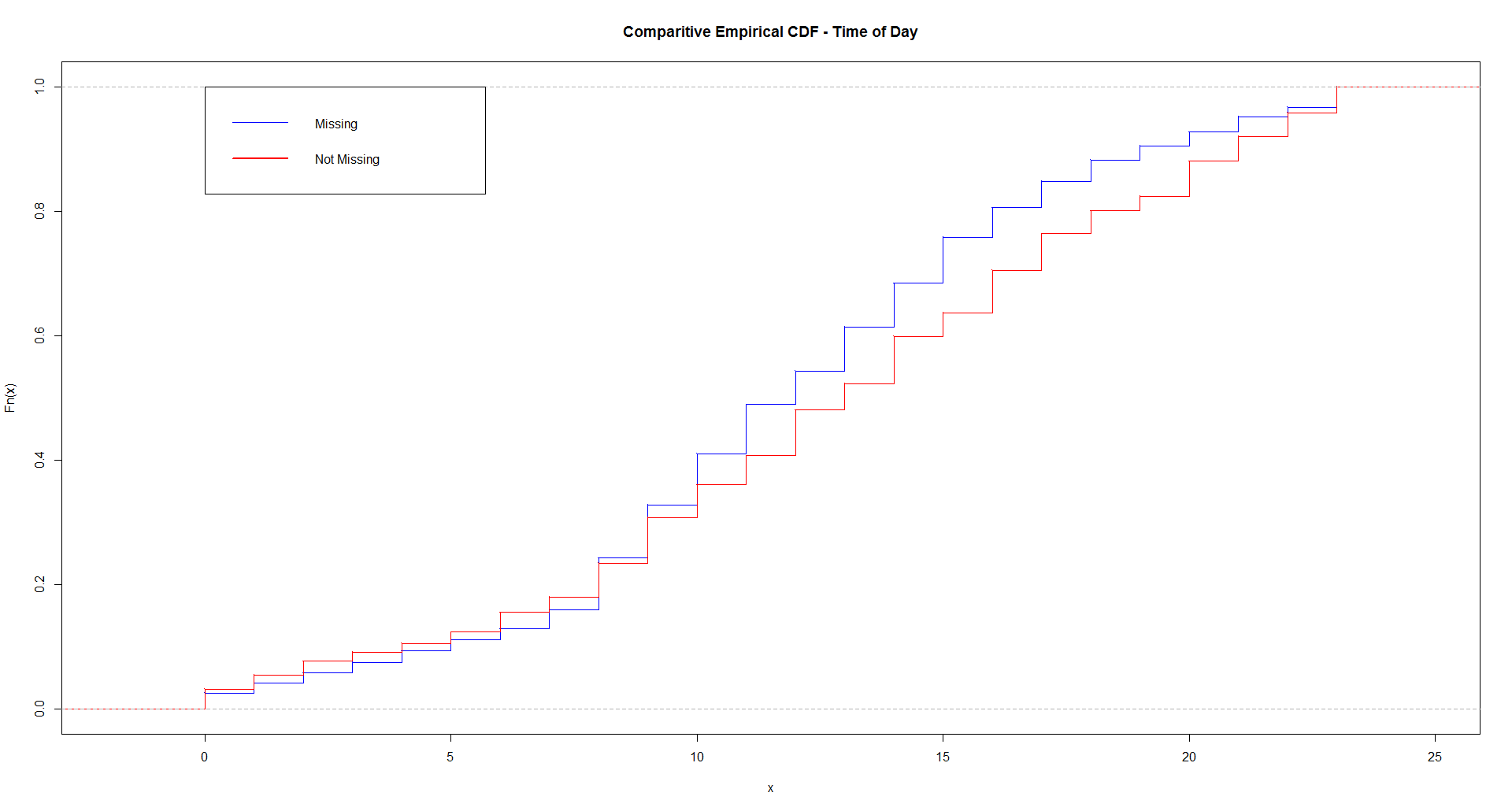
If we filter the rows where the variable “spampct” is present, we are left with 818 rows.

The correlation matrix is presented below. We can see that there is a correlation of 0.71 between “spam” and “spampct”.

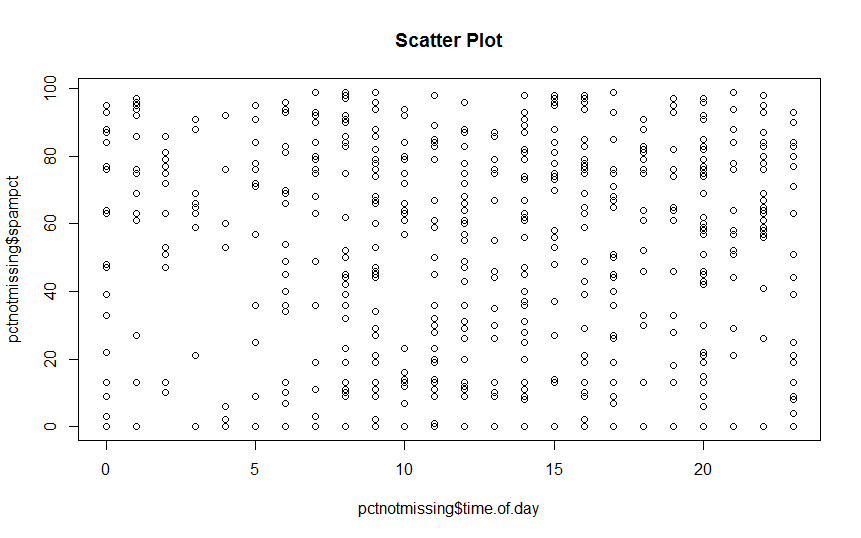


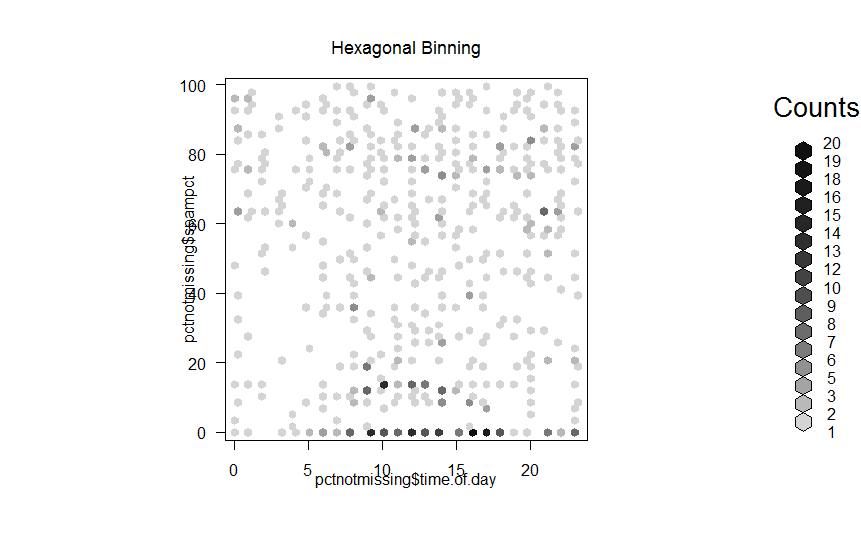
1. There are 1.353 values missing, so we have 818 rows with valid values.
2. There are some differences of the variable “time.of.day” when the variable “spampct” is present versus when it is missing.
   1. The average when “spampct” is missing is 11.95 when it is present, 12.78
   2. The 3rd quartile is 15 vs 17. The rest of the quartiles are similar. This means that most of the 1.353 values are between 15 and 17 hours.





1. We can use hexagonal binning graphs to deal with the overplotting and get a better representation than the scatter plot. We can see that there is an important concentration of points along the line with 0 value for the column “spampct”





Question 3. Read and did Lab

Question 4. Exercise 13 in Chapter 3 of ISLR.

1. Creating 100 random numbers for x.

> set.seed(1)

> x <- rnorm(100)

> x

[1] -0.626453811 0.183643324 -0.835628612 1.595280802 0.329507772 -0.820468384 0.487429052 0.738324705 0.575781352

[10] -0.305388387 1.511781168 0.389843236 -0.621240581 -2.214699887 1.124930918 -0.044933609 -0.016190263 0.943836211

[19] 0.821221195 0.593901321 0.918977372 0.782136301 0.074564983 -1.989351696 0.619825748 -0.056128740 -0.155795507

[28] -1.470752384 -0.478150055 0.417941560 1.358679552 -0.102787727 0.387671612 -0.053805041 -1.377059557 -0.414994563

[37] -0.394289954 -0.059313397 1.100025372 0.763175748 -0.164523596 -0.253361680 0.696963375 0.556663199 -0.688755695

[46] -0.707495157 0.364581962 0.768532925 -0.112346212 0.881107726 0.398105880 -0.612026393 0.341119691 -1.129363096

[55] 1.433023702 1.980399899 -0.367221476 -1.044134626 0.569719627 -0.135054604 2.401617761 -0.039240003 0.689739362

[64] 0.028002159 -0.743273209 0.188792300 -1.804958629 1.465554862 0.153253338 2.172611670 0.475509529 -0.709946431

[73] 0.610726353 -0.934097632 -1.253633400 0.291446236 -0.443291873 0.001105352 0.074341324 -0.589520946 -0.568668733

[82] -0.135178615 1.178086997 -1.523566800 0.593946188 0.332950371 1.063099837 -0.304183924 0.370018810 0.267098791

[91] -0.542520031 1.207867806 1.160402616 0.700213650 1.586833455 0.558486426 -1.276592208 -0.573265414 -1.224612615

[100] -0.473400636

1. Creating 100 random numbers for eps, with mean 0 and variance 0.25 (standard deviation )

> eps <- rnorm(100, mean=0, sd=sqrt(0.25))

> eps

[1] -0.310183339 0.021057937 -0.455460824 0.079014386 -0.327292322 0.883643635 0.358353738 0.455087115 0.192092679

[10] 0.841088040 -0.317868227 -0.230822365 0.716141119 -0.325348177 -0.103690372 -0.196403965 -0.159996434 -0.139556651

[19] 0.247094166 -0.088665241 -0.252978731 0.671519413 -0.107289704 -0.089778265 -0.050095371 0.356333154 -0.036782202

[28] -0.018817086 -0.340830239 -0.162135136 0.030080220 -0.294447243 0.265748096 -0.759197041 0.153278930 -0.768224912

[37] -0.150488063 -0.264139952 -0.326047390 -0.028448389 -0.957179713 0.588291656 -0.832486218 -0.231765201 -0.557960053

[46] -0.375409501 1.043583273 0.008697810 -0.643150265 -0.820302767 0.225093551 -0.009279916 -0.159034187 -0.464681074

[55] -0.743730155 -0.537596148 0.500014402 -0.310633347 -0.692213424 0.934645311 0.212550189 -0.119323550 0.529241524

[64] 0.443211326 -0.309621524 1.103051232 -0.127513515 -0.712247325 -0.072199801 0.103769170 1.153989200 0.052901184

[73] 0.228499403 -0.038576468 -0.167000421 -0.017363014 0.393819803 1.037622504 0.513696219 0.603954199 -0.615661711

[82] 0.491947785 0.109962402 -0.733625015 0.260511371 -0.079377302 0.732293656 -0.383041000 -0.215105877 -0.463054749

[91] -0.088551981 0.201005890 -0.365874087 0.415186584 -0.604041393 -0.523992206 0.720578853 -0.507923733 0.205987356

[100] -0.190538026

1. Generating vector y. Length of y is 100. ,

y <- -1 + 0.5\*x+eps

> y

[1] -1.62341024 -0.88712040 -1.87327513 -0.12334521 -1.16253844 -0.52659056 -0.39793174 -0.17575053 -0.52001665

[10] -0.31160615 -0.56197764 -1.03590075 -0.59447917 -2.43269812 -0.54122491 -1.21887077 -1.16809157 -0.66763855

[19] -0.34229524 -0.79171458 -0.79349005 0.06258756 -1.07000721 -2.08445411 -0.74018250 -0.67173122 -1.11467996

[28] -1.75419328 -1.57990527 -0.95316436 -0.29058000 -1.34584111 -0.54041610 -1.78609956 -1.53525085 -1.97572219

[37] -1.34763304 -1.29379665 -0.77603470 -0.64686051 -2.03944151 -0.53838918 -1.48400453 -0.95343360 -1.90233790

[46] -1.72915708 0.22587425 -0.60703573 -1.69932337 -1.37974890 -0.57585351 -1.31529311 -0.98847434 -2.02936262

[55] -1.02721830 -0.54739620 -0.68359634 -1.83270066 -1.40735361 -0.13288199 0.41335907 -1.13894355 -0.12588879

[64] -0.54278759 -1.68125813 0.19744738 -2.02999283 -0.97946989 -0.99557313 0.19007500 0.39174396 -1.30207203

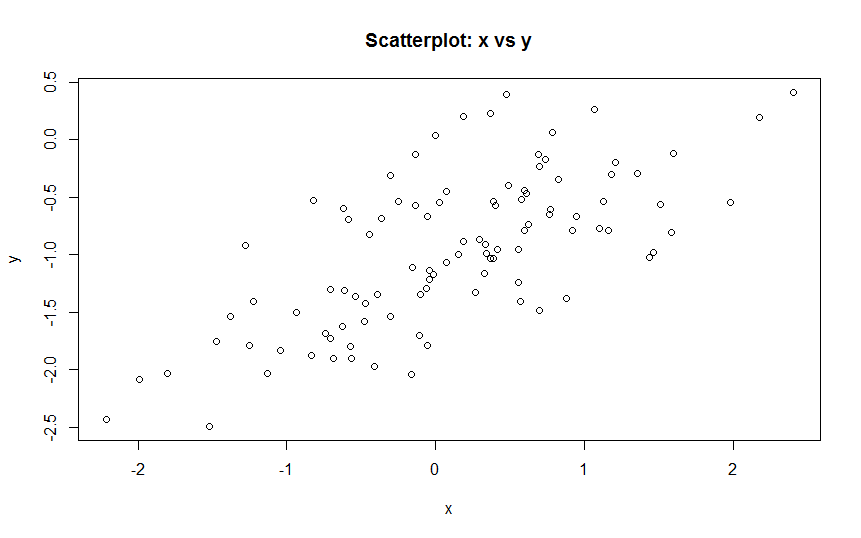
[73] -0.46613742 -1.50562528 -1.79381712 -0.87163990 -0.82782613 0.03817518 -0.44913312 -0.69080627 -1.89999608

[82] -0.57564152 -0.30099410 -2.49540841 -0.44251553 -0.91290212 0.26384357 -1.53513296 -1.03009647 -1.32950535

[91] -1.35981200 -0.19506021 -0.78567278 -0.23470659 -0.81062467 -1.24474899 -0.91771725 -1.79455644 -1.40631895

[100] -1.42723834

1. In the scatterplot below we can recognize several things,
   1. The range of x is approximately from -2.2 to 2.2
   2. The range of y is approximately from -2.5 to 0.4
   3. As x grows, so does y, there is a positive correlation between them.
   4. Most of the points are lying inside the box



1. Fitting a linear model to relate y to x, we see

, 2% greater than .

, 0.2% smaller than

lm.fit=lm(y~x)

> summary(lm.fit)

Call:

lm(formula = y ~ x)

Residuals:

Min 1Q Median 3Q Max

-0.93842 -0.30688 -0.06975 0.26970 1.17309

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -1.01885 0.04849 -21.010 < 2e-16 \*\*\*

x 0.49947 0.05386 9.273 4.58e-15 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4814 on 98 degrees of freedom

Multiple R-squared: 0.4674, Adjusted R-squared: 0.4619

F-statistic: 85.99 on 1 and 98 DF, p-value: 4.583e-15

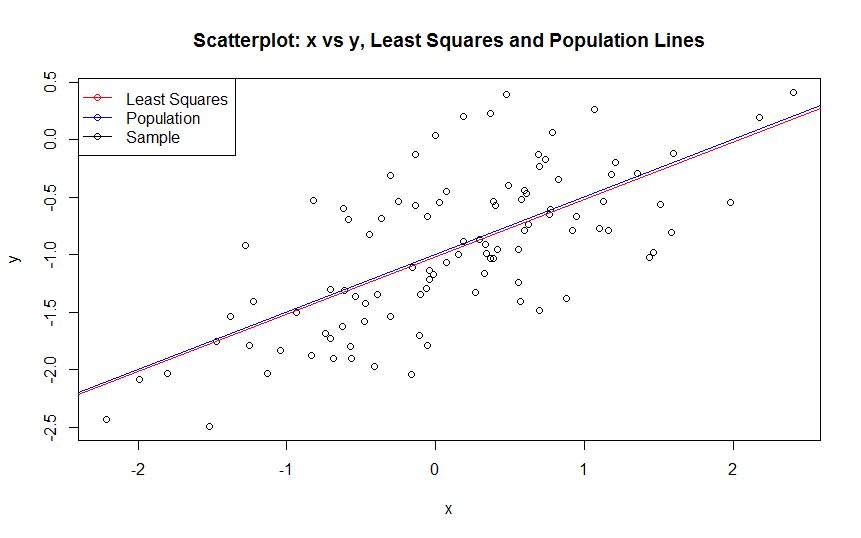
1. As we can see in the scatter plot, the least squares line and the population line are quite close, very parallel, and go through the middle of the points.

plot(x,y, title(main="Scatterplot: x vs y, Least Squares and Population Lines") )

lineLM <- abline(lm(y~x), col="red")

linePop <- abline(a=-1, b=0.5, col="blue")

legend("topleft", legend = c("Least Squares", "Population", "Sample"), lwd=1, pch = 1, col=c("red", "blue", "black"))



1. When we compare the model with and without the quadratic value, we can see that there is NO evidence that the quadratic term improves the model fit, because
   1. the value is essentially the same when we add the quadratic term (0.4619 vs 0.4672)
   2. The p-value for the coefficient of is  pvalue(, suggesting that we do not reject the null hypothesis that

|  |  |
| --- | --- |
| Model without quadratic term  Call:  lm(formula = y ~ x)  Residuals:  Min 1Q Median 3Q Max  -0.93842 -0.30688 -0.06975 0.26970 1.17309  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) -1.01885 0.04849 -21.010 < 2e-16 \*\*\*  x 0.49947 0.05386 9.273 4.58e-15 \*\*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 0.4814 on 98 degrees of freedom  **Multiple R-squared: 0.4674, Adjusted R-squared: 0.4619**  F-statistic: 85.99 on 1 and 98 DF, p-value: 4.583e-15 | Model with quadratic term  Call:  lm(formula = y ~ x + I(x^2))  Residuals:  Min 1Q Median 3Q Max  -0.98252 -0.31270 -0.06441 0.29014 1.13500  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) -0.97164 0.05883 -16.517 < 2e-16 \*\*\*  x 0.50858 0.05399 9.420 2.4e-15 \*\*\*  I(x^2) -0.05946 0.04238 -1.403 0.164  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 0.479 on 97 degrees of freedom  **Multiple R-squared: 0.4779, Adjusted R-squared: 0.4672**  F-statistic: 44.4 on 2 and 97 DF, p-value: 2.038e-14 |

1. A new dataset was created with **less** noise, having

As a general sense, we can see that the least squares model fits the data better, because the statistic is greater than before.

When we compare the model with and without the quadratic value, we can see that there is NO evidence that the quadratic term improves the model fit, because

* 1. the value is essentially the same when we add the quadratic term (0.6838 vs 0.6869)
  2. The p-value for the coefficient of is  pvalue(, suggesting that we do not reject the null hypothesis that
     1. The p-value is the same as the original model

|  |  |
| --- | --- |
| Model without quadratic term  Call:  lm(formula = y ~ x)  Residuals:  Min 1Q Median 3Q Max  -0.59351 -0.19409 -0.04411 0.17057 0.74193  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) -1.01192 0.03067 -32.99 <2e-16 \*\*\*  x 0.49966 0.03407 14.67 <2e-16 \*\*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 0.3044 on 98 degrees of freedom  **Multiple R-squared: 0.687, Adjusted R-squared: 0.6838**  F-statistic: 215.1 on 1 and 98 DF, p-value: < 2.2e-16 | Model with quadratic term  Call:  lm(formula = y ~ x + I(x^2))  Residuals:  Min 1Q Median 3Q Max  -0.62140 -0.19777 -0.04073 0.18350 0.71783  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) -0.98207 0.03721 -26.395 <2e-16 \*\*\*  x 0.50543 0.03415 14.801 <2e-16 \*\*\*  I(x^2) -0.03761 0.02681 -1.403 0.164  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 0.303 on 97 degrees of freedom  **Multiple R-squared: 0.6933, Adjusted R-squared: 0.6869**  F-statistic: 109.6 on 2 and 97 DF, p-value: < 2.2e-16 |

1. A new dataset was created with **more** noise, having

As a general sense, we can see that the least squares model the data worse than before, because the statistic is lower than before.

When we compare the model with and without the quadratic value, we can see that there is NO evidence that the quadratic term improves the model fit, because

* 1. the value is essentially the same when we add the quadratic term (0.1712 vs 0.1793)
  2. The p-value for the coefficient of is  pvalue(, suggesting that we do not reject the null hypothesis that
     1. The p-value is the same as both models with less noise.

|  |  |
| --- | --- |
| Model without quadratic term  Call:  lm(formula = y ~ x)  Residuals:  Min 1Q Median 3Q Max  -1.8768 -0.6138 -0.1395 0.5394 2.3462  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) -1.03769 0.09699 -10.699 < 2e-16 \*\*\*  x 0.49894 0.10773 4.632 1.12e-05 \*\*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 0.9628 on 98 degrees of freedom  Multiple R-squared: 0.1796, Adjusted R-squared: 0.1712  F-statistic: 21.45 on 1 and 98 DF, p-value: 1.117e-05 | Model with quadratic term  Call:  lm(formula = y ~ x + I(x^2))  Residuals:  Min 1Q Median 3Q Max  -1.9650 -0.6254 -0.1288 0.5803 2.2700  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) -0.94328 0.11766 -8.017 2.47e-12 \*\*\*  x 0.51716 0.10798 4.789 6.01e-06 \*\*\*  I(x^2) -0.11892 0.08477 -1.403 0.164  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 0.958 on 97 degrees of freedom  Multiple R-squared: 0.1959, Adjusted R-squared: 0.1793  F-statistic: 11.82 on 2 and 97 DF, p-value: 2.557e-05 |

1. The confidence intervals of the model coefficients , for each level of noise (variance of ) are displayed below,

We can comment that the size of the confidence interval grows as the data has more noise.

0.1

2.5 % 97.5 % **Size**

(Intercept) -1.0727832 -0.9510557 **0.12**

x 0.4320613 0.5672681 **0.13**

0.25

2.5 % 97.5 % **Size**

(Intercept) -1.1150804 -0.9226122 **0.19**

x 0.3925794 0.6063602 **0.22**

1.0

2.5 % 97.5 % **Size**

(Intercept) -1.2301607 -0.8452245 **0.385**

x 0.2851588 0.7127204 **0.427**

Question 5.

1. The regression for each case of missing data are posted below, as well as the graph of the best fit line for each of the four cases. The main take aways are:
   1. MCAR and MAR produce slopes similar than the case with no missing data. A couple of remarks:
      1. MCAR underestimates the slope
      2. MAR over estimates the slope. Making sense because we kept only the Month 2 values above 140.
   2. MNAR produces an estimate that is definitively different from the other three. The slope is near zero, and even the intercept is totally different.

No missing data,

Call:

lm(formula = values[, 2] ~ values[, 1])

Residuals:

Min 1Q Median 3Q Max

-64.735 -15.606 0.244 16.183 49.376

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 108.84344 17.49684 6.221 1.15e-07 \*\*\*

values[, 1] 0.07807 0.13911 0.561 0.577

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 23.9 on 48 degrees of freedom

Multiple R-squared: 0.006519, Adjusted R-squared: -0.01418

F-statistic: 0.315 on 1 and 48 DF, p-value: 0.5773

MCAR

Call:

lm(formula = mcar[, 2] ~ mcar[, 1])

Residuals:

Min 1Q Median 3Q Max

-28.513 -10.394 2.964 5.656 39.535

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 52.8698 23.9125 2.211 0.0472 \*

mcar[, 1] 0.5314 0.1895 2.804 0.0159 \*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 19.97 on 12 degrees of freedom

Multiple R-squared: 0.3958, Adjusted R-squared: 0.3455

F-statistic: 7.862 on 1 and 12 DF, p-value: 0.01593

MAR

Call:

lm(formula = mar[, 2] ~ mar[, 1])

Residuals:

Min 1Q Median 3Q Max

-22.430 -7.875 1.441 7.459 24.046

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 25.7068 46.8712 0.548 0.5934

mar[, 1] 0.7427 0.2953 2.515 0.0271 \*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.25 on 12 degrees of freedom

Multiple R-squared: 0.3452, Adjusted R-squared: 0.2906

F-statistic: 6.326 on 1 and 12 DF, p-value: 0.02715

MNAR

Call:

lm(formula = mnar[, 2] ~ mnar[, 1])

Residuals:

Min 1Q Median 3Q Max

-11.1252 -4.0594 0.6142 3.9821 9.3616

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 141.48302 12.87575 10.988 6.66e-07 \*\*\*

mnar[, 1] 0.08379 0.08352 1.003 0.339

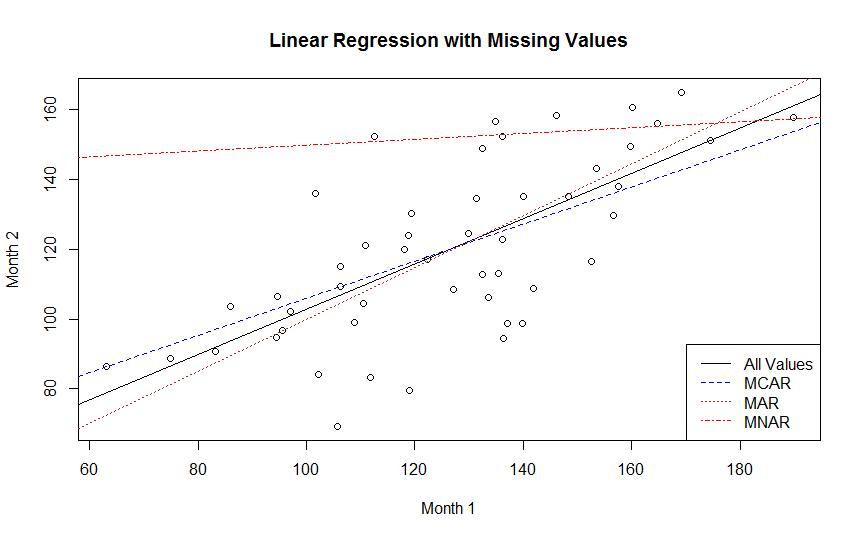
---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.924 on 10 degrees of freedom

Multiple R-squared: 0.09144, Adjusted R-squared: 0.000581

F-statistic: 1.006 on 1 and 10 DF, p-value: 0.3394



1. The table below displays the average estimate for , we can see that
   1. MCAR has a very similar result as all data.
   2. MAR has some similitude to all data, but less similar than MCAR.
   3. MNAR is very far away from the coefficient of all data.

Scenario Intercept Slope

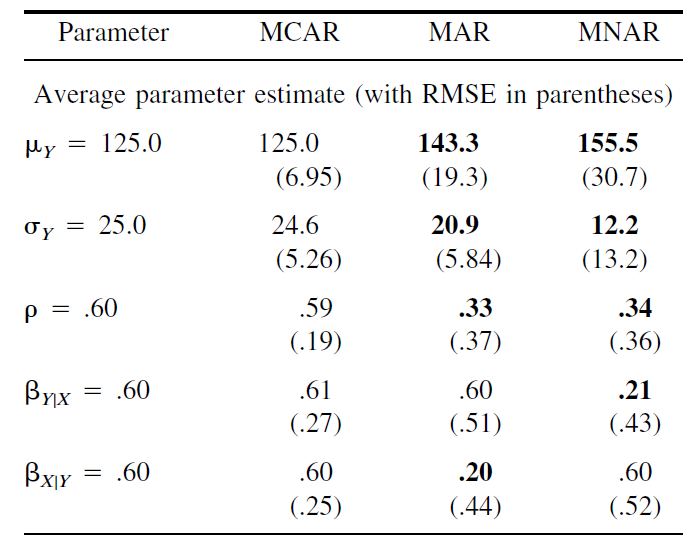
[1,] "all data" "50.3619363864206" "0.596893638989437"

[2,] "MCAR" "49.7455552732067" "0.60217642375175"

[3,] "MAR" "47.9984998072741" "0.611931500031285"

[4,] "MNAR" "126.743283806805" "0.19946433034383"

1. Table 2 of SchaferGraham is displayed below, we will check for and for



**To calculate** , we use the formula .

The value of , is calculated using all the values of Month 1.

The betas for each MCAR, ,MAR, MNAR is the result of the average across all regressions, across the T=100 samples.

Looking at the Betas, we can see that MCAR has the closest average value, followed by MAR, and MNAR is clearly very far.

> c(muY.mcar, muY.mar, muY.mnar)

[1] 122.5954 122.0285 150.8740

**For ,** we must look at , and we find a similar conclusion as before. MCAR is closest to the population, followed by MAR and MNAR is clearly very far.

Scenario Slope

[2,] "MCAR" "0.60217642375175"

[3,] "MAR" "0.611931500031285"

[4,] "MNAR" "0.19946433034383"

**Comparing to the paper**, we can see in the paper MCAR was offering estimations closest to the population, as well as in this study.

Problem 6.

1. The least squares estimator for can be derived as follows

We derive with respect to variable “c” and equal to zero to find the minimizer,



(ii) We compute the bias squared, using the result from part (i)

(iii) Computing the variance

1. With the new estimator, the error can be expressed as,

Computing first ,

Computing first the bias squared,

Computing variance,

To minimize we derive with respect to beta and equal to zero,

We equal to zero and solve for ,