BACS HW (Week 9)

108020024

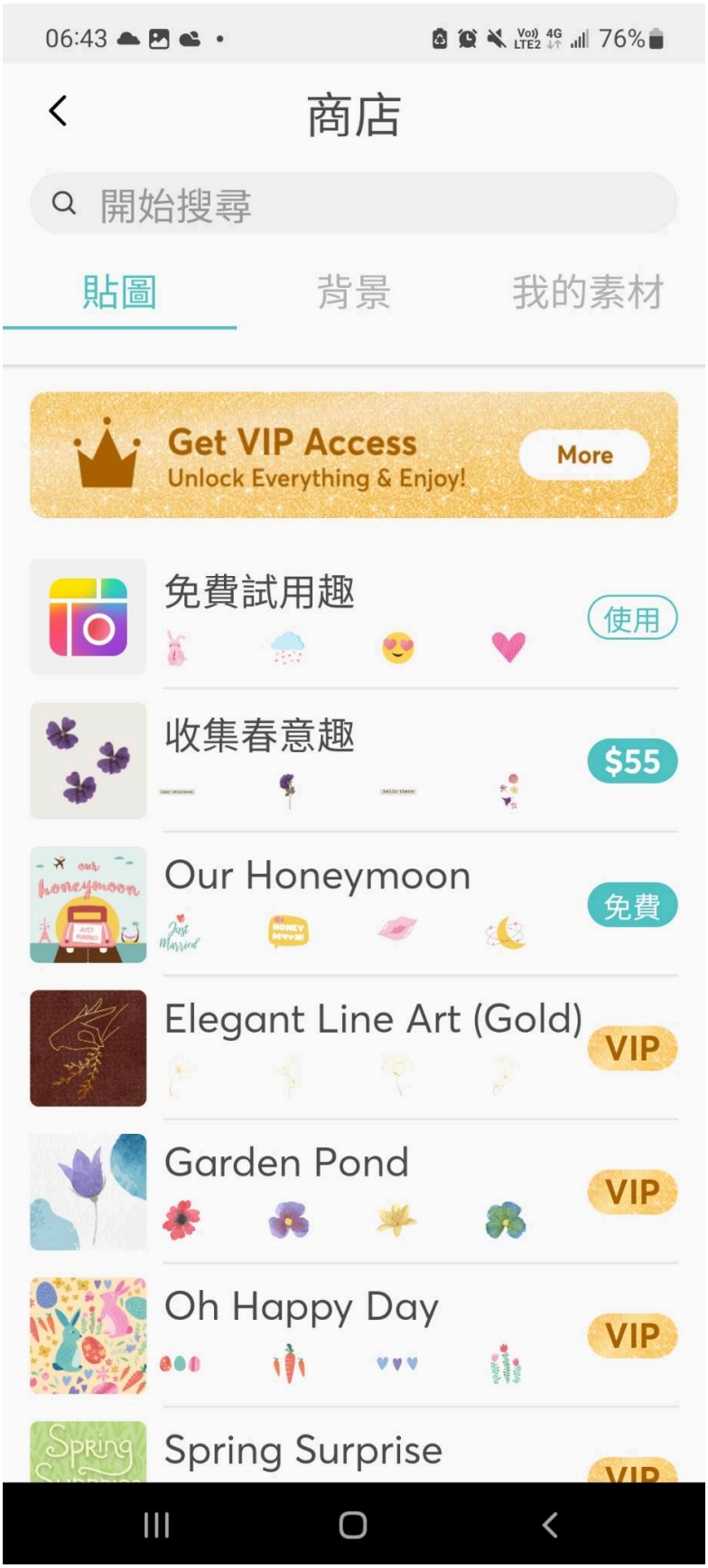
due on 04/16 (Sun) Helped by 108020033

Question 1) Let's explore and describe the data and develop some early intuitive thoughts:

```
library(data.table)
ac_bundles_dt <- fread("piccollage_accounts_bundles.csv")
ac_bundles_matrix <- as.matrix(ac_bundles_dt[, -1, with=FALSE])
```

a)Let's explore to see if any sticker bundles seem intuitively similar:

i) (recommended) Download PicCollage onto your mobile from the App Store and take a look at the styl


06:43
76%
商店
開始搜尋
貼圖
背景
我的素材
Get VIP Access
Unlock Everything & Enjoy!
More
免費試用趣
使用
收集春意趣
$55
Our Honeymoon
免費
Elegant Line Art (Gold)
VIP
Garden Pond
VIP
Oh Happy Day
VIP
Spring Surprise
VIP

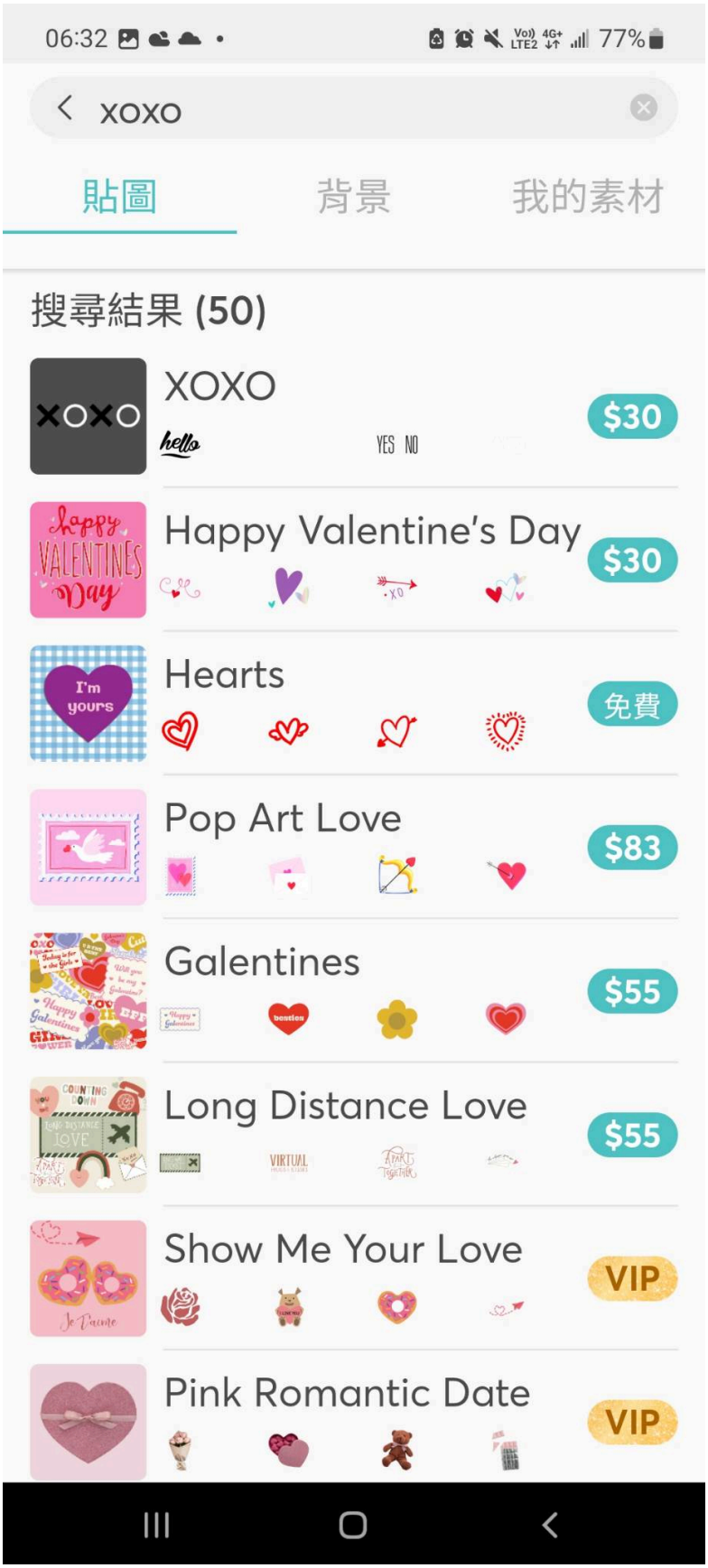
ii) Find a single sticker bundle that is both in our limited data set and also in the app's Sticker

```
colnames(ac_bundles_dt)
```

```
##   [1] "account_id"               "Maroon5V"
##   [3] "between"                  "pellington"
##   [5] "StickerLite"              "saintvalentine"
##   [7] "HipsterChicSara"          "OddAnatomy"
##   [9] "wonderland"               "V10"
##  [11] "lovestinks2016"           "Random"
##  [13] "supercute"                "retrosummer"
##  [15] "Emome"                    "toMomwithLove"
##  [17] "thebouqs"                 "HeartStickerPack"
##  [19] "bubbleletters"            "gwen"
##  [21] "food"                     "Monsterhigh"
##  [23] "supersweet"               "word"
##  [25] "xmasquotes"               "WinterWonderland"
##  [27] "fallinlovewiththefall"    "alien"
##  [29] "nashnext"                 "carfriends"
##  [31] "peanutmangif"             "KLL"
##  [33] "CutieV"                   "snowflakes"
##  [35] "newyearsparty"            "betweenspring"
##  [37] "simplyautumn"             "beatsmusic"
##  [39] "xoxo"                     "togetherwerise"
##  [41] "RageComics"               "chubbles"
##  [43] "happycny2016"             "happy"
##  [45] "seaamo"                   "hellobaby"
##  [47] "doodlewords"              "chloecaroline"
##  [49] "bmnemesis"                "arrows"
##  [51] "hbd2016"                  "freshempire"
##  [53] "Mom2013"                  "stpatrick"
##  [55] "icecreamsocial"           "sassyhween"
##  [57] "sphalloween"              "CatpixCubie"
##  [59] "kungfood"                 "eastersurprise"
##  [61] "happyeaster2016"          "frombierun"
##  [63] "DecktheHall"              "Eggotown"
##  [65] "papertapes"               "Valentine2013StickerPack"
##  [67] "peanutman"                "Dad2013"
##  [69] "PhotoboothFest"           "WherezSanta"
##  [71] "bananaman"                "Halloween2012StickerPack"
##  [73] "mmlm"                     "julyfourth"
##  [75] "tropicalparadise"         "bestdaddy"
##  [77] "sweetmothersday"          "springrose"
##  [79] "wpbear"                   "autumn"
##  [81] "justmytype"               "gudetama"
##  [83] "backtocool"               "8bit2"
##  [85] "4thofjuly3"               "summerlovin"
##  [87] "superherodad2"            "hipsteroverlays"
##  [89] "watercolor"               "hellospring"
##  [91] "supersassy"               "cutoutluv"
##  [93] "ladolcevita"              "bemine"
##  [95] "japan2015"                "doodleholiday"
```

```
##  [97] "washiholiday"             "hipsterholiday"
##  [99] "creepycute"               "2014summer"
## [101] "cherngs"                  "vintage"
## [103] "AntiV"                    "BlingStickerPack"
## [105] "HalloweenScream2013"      "GraffitiStickerPack2013"
## [107] "RobinThicke2013"          "AnimalFriendsStickerPack"
## [109] "HipsterChic"              "PartyStickerPack"
## [111] "Music1D"                  "CampusLife"
## [113] "graduation2015"           "aroundtheworld"
## [115] "WordsStickerPack"         "NaiveLittleThings"
## [117] "Holiday2012StickerPack"   "jollyholiday"
## [119] "valentineStickers"        "Xmas2012StickerPack"
## [121] "alphabet"                 "forever"
## [123] "toyoufromme"              "AccessoriesStickerPack"
## [125] "fifacomics"               "2014fifa"
## [127] "StampStickerPack"         "ouija"
## [129] "stationery"               "ChineseNewYear2013"
## [131] "sanrio"                   "family"
## [133] "babyanimals"              "halloweenparty"
## [135] "costumeparty"             "starrytribe"
## [137] "cutevalentine"            "holidaycheers"
## [139] "givethanks"               "teenwitch"
## [141] "mrcurlsport"              "vote2016"
## [143] "floralwedding"            "happybday"
## [145] "chicchristmas"            "snowflakeee"
## [147] "hkbts"                    "warmncozy"
## [149] "aesthetics"               "christmassnow"
## [151] "pacmanholiday"            "vintagexmas"
## [153] "yummyfood"                "watercolorywinter"
## [155] "wordstoliveby"            "helloautumn"
## [157] "dayofdead"                "summergetaway"
## [159] "gradparty"                "xmassketches"
## [161] "cometobe"                 "glitterny"
## [163] "vintagewashi"             "notetoself"
## [165] "salelabels"               "cny2017"
```

Choose **xoxo** as example.


06:32
77%
xoxo
貼圖
背景
我的素材
搜尋結果 (50)
XOXO
hello
YES NO
$30
Happy Valentine's Day
$30
Hearts
免費
Pop Art Love
$83
Galentines
$55
Long Distance Love
$55
Show Me Your Love
VIP
Pink Romantic Date
VIP

Guess: 1.CutieV 2.valentineStickers 3.HeartStickerPack 4.supersweet 5.cutevalentine may also have similar usage patterns as this bundle.

b) Let's find similar bundles using geometric models of similarity:

i) Let's create cosine similarity based recommendations for all bundles:

1.)Create a matrix or data.frame of the top 5 recommendations for all bundles

2.)Create a new function that automates the above functionality: it should take an accounts

```
#install.packages("lsa")
library(lsa)
```

```
##     SnowballC
```

```
sort_data <- function(x){
  top6=x[order(x, decreasing=T)[2:6]]
  attributes(top6)$names
}
### function contribution to


recommendations=t(apply(cosine(ac_bundles_matrix), 1, sort_data))
head(recommendations)
```

```
##                 [,1]               [,2]               [,3]
## Maroon5V        "OddAnatomy"       "beatsmusic"       "xoxo"
## between         "BlingStickerPack" "xoxo"             "gwen"
## pellington      "springrose"       "8bit2"            "mmlm"
## StickerLite     "HeartStickerPack" "HipsterChicSara"  "Mom2013"
## saintvalentine  "nashnext"         "givethanks"       "teenwitch"
## HipsterChicSara "Random"           "HeartStickerPack" "wonderland"
##                 [,4]             [,5]
## Maroon5V        "alien"          "word"
## between         "OddAnatomy"     "AccessoriesStickerPack"
## pellington      "julyfourth"     "tropicalparadise"
## StickerLite     "Emome"          "Random"
## saintvalentine  "togetherwerise" "lovestinks2016"
## HipsterChicSara "Emome"          "StickerLite"
```

3.)What are the top 5 recommendations for the bundle you chose to explore earlier?

```
recommendations["xoxo",]
```

```
## [1] "BlingStickerPack" "OddAnatomy"       "between"          "gwen"
## [5] "KLL"
```

The top 5 recommendations for the bundle is:

1."BlingStickerPack" 2."OddAnatomy" 3."between" 4."gwen" 5."KLL"

ii) Let's create correlation based recommendations.

1.)Reuse the function you created above (don't change it; don't use the cor() function)

2.)But this time give the function an accounts-bundles matrix where each bundle (column) ha

```
sort_data <- function(x){
  top6=x[order(x, decreasing=T)[2:6]]
  attributes(top6)$names
}
### function contribution to


recommendations=t(apply(cosine(scale(ac_bundles_matrix, scale = FALSE)), 1, sort_data))
head(recommendations)
```

```
##                 [,1]               [,2]
## Maroon5V        "OddAnatomy"       "beatsmusic"
## between         "BlingStickerPack" "xoxo"
## pellington      "springrose"       "8bit2"
## StickerLite     "HeartStickerPack" "AnimalFriendsStickerPack"
## saintvalentine  "nashnext"         "givethanks"
## HipsterChicSara "Random"           "HeartStickerPack"
##                 [,3]               [,4]              [,5]
## Maroon5V        "xoxo"             "alien"           "word"
## between         "gwen"             "OddAnatomy"      "AccessoriesStickerPack"
## pellington      "tropicalparadise" "mmlm"            "julyfourth"
## StickerLite     "between"          "Emome"           "HipsterChicSara"
## saintvalentine  "teenwitch"        "togetherwerise"  "lovestinks2016"
## HipsterChicSara "wonderland"       "Emome"           "StickerLite"
```

3.)Now what are the top 5 recommendations for the bundle you chose to explore earlier?

```
recommendations["xoxo",]
```

```
## [1] "BlingStickerPack" "OddAnatomy"       "between"          "gwen"
## [5] "KLL"
```

The top 5 recommendations for the bundle is:

1."BlingStickerPack" 2."OddAnatomy" 3."between" 4."gwen" 5."KLL"

iii) Let's create adjusted-cosine based recommendations.

1.)Reuse the function you created above (you should not have to change it)

2.)But this time give the function an accounts-bundles matrix where each account (row) has

```
sort_data <- function(x){
  top6=x[order(x, decreasing=T)[2:6]]
  attributes(top6)$names
}
### function contribution to


recommendations=t(apply(cosine(ac_bundles_matrix-rowMeans(ac_bundles_matrix)), 1, sort_data))
head(recommendations)
```

```
##                 [,1]               [,2]               [,3]
## Maroon5V        "OddAnatomy"       "word"             "xoxo"
## between         "BlingStickerPack" "xoxo"             "gwen"
## pellington      "springrose"       "8bit2"            "backtocool"
## StickerLite     "HeartStickerPack" "Mom2013"          "HipsterChicSara"
## saintvalentine  "togetherwerise"   "givethanks"       "teenwitch"
## HipsterChicSara "Random"           "HeartStickerPack" "wonderland"
##                 [,4]               [,5]
## Maroon5V        "beatsmusic"       "supercute"
## between         "Monsterhigh"      "OddAnatomy"
## pellington      "tropicalparadise" "julyfourth"
## StickerLite     "Emome"            "Random"
## saintvalentine  "mrcurlsport"      "arrows"
## HipsterChicSara "Emome"            "StickerLite"
```

3.)What are the top 5 recommendations for the bundle you chose to explore earlier?

```
recommendations["xoxo",]
```

```
## [1] "BlingStickerPack" "between"          "OddAnatomy"       "gwen"
## [5] "Monsterhigh"
```

The top 5 recommendations for the bundle is:

1."BlingStickerPack" 2."between" 3."OddAnatomy" 4."gwen" 5."Monsterhigh"

iii) (not graded) Are the three sets of geometric recommendations similar in nature (theme/keywords)

No, I thought xoxo will be related with love or valentines, I really don't know the reason to explain the different recommendation.

iv) (not graded) What do you think is the conceptual difference in cosine similarity, correlation, a

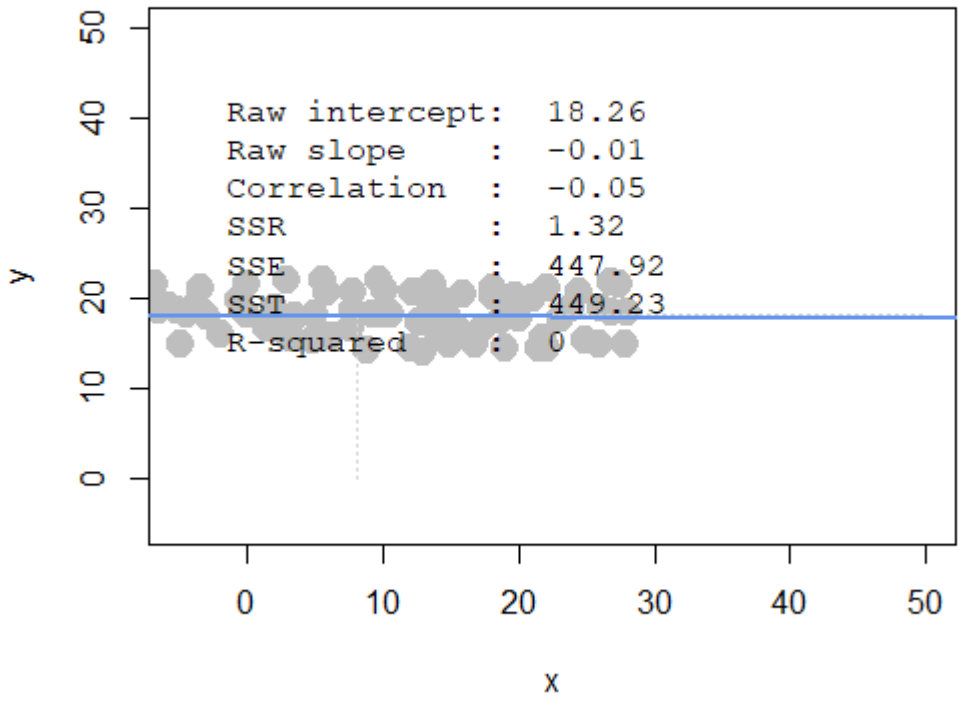
https://www.researchgate.net/post/Can_someone_differentiate_between_Cosine_Adjusted_cosine_and_Pearson_correlation_similarity_measuring_techniques

Mr. Alexander Egoyan gives a clean explanation about it.

Question 2) Correlation is at the heart of many data analytic methods so let's explore it further.

```
library(compstatslib)
```

a) Scenario A: Create a horizontal set of random points, with a relatively narrow but flat distribution.



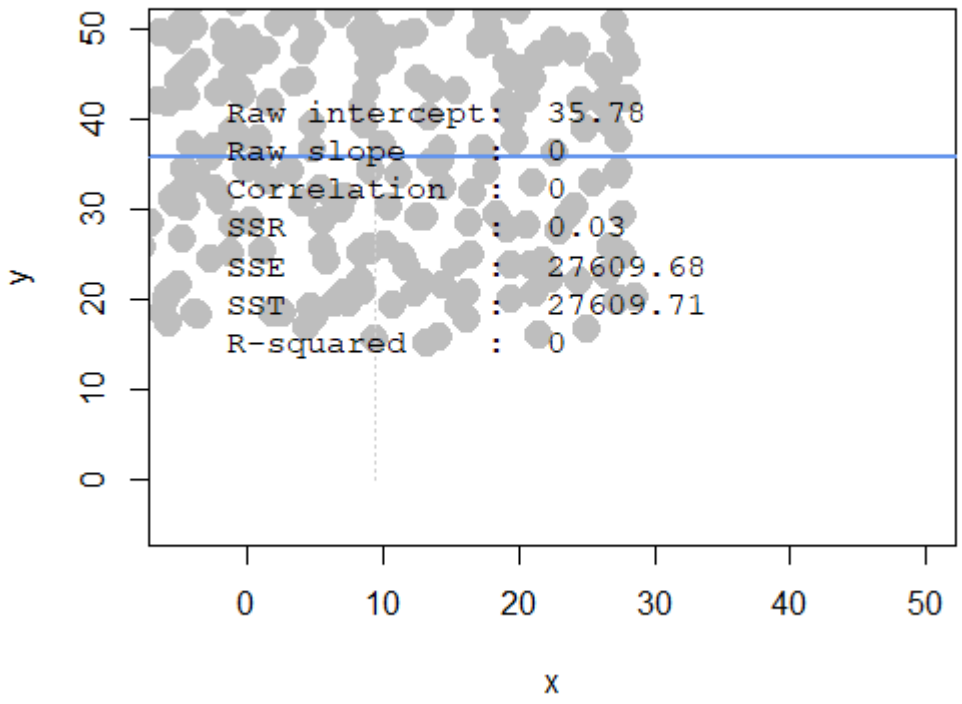
i) What raw slope of x and y would you generally expect?

A: 0

ii) What is the correlation of x and y that you would generally expect?

A: 0

b) Scenario B: Create a random set of points to fill the entire plotting area, along both x-axis and y-axis



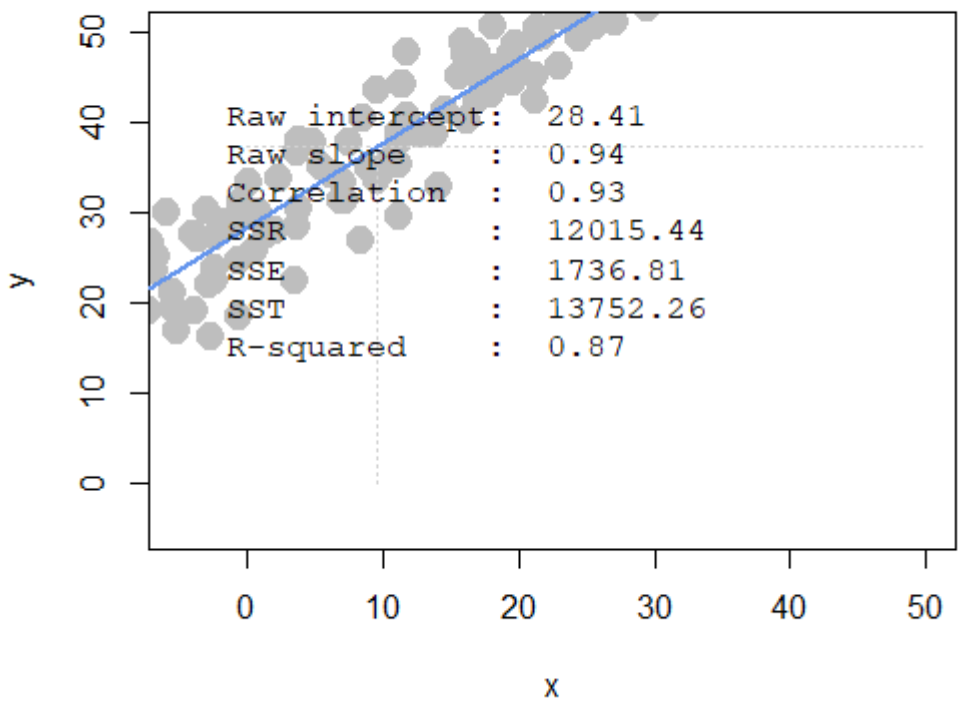
i) What raw slope of x and y would you generally expect?

A: 0

ii) What is the correlation of x and y that you would generally expect?

A: 0

c) Scenario C: Create a diagonal set of random points trending upwards at 45 degrees



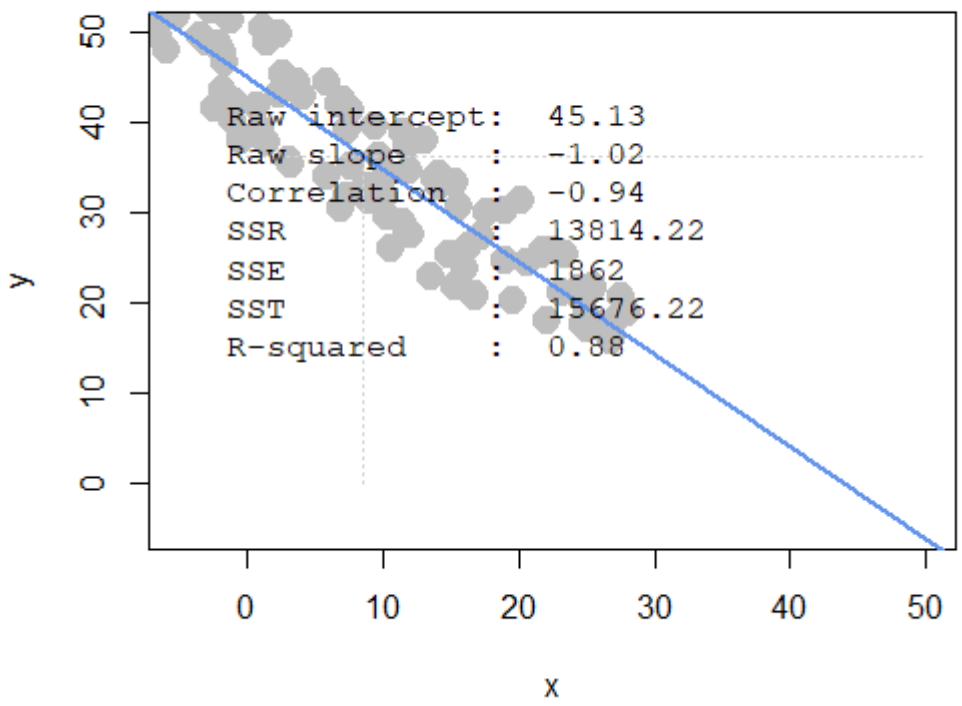
i) What raw slope of x and y would you generally expect?

A: 1

ii) What is the correlation of x and y that you would generally expect?

A: 1

d) Scenario D: Create a diagonal set of random trending downwards at 45 degrees



i) What raw slope of x and y would you generally expect?

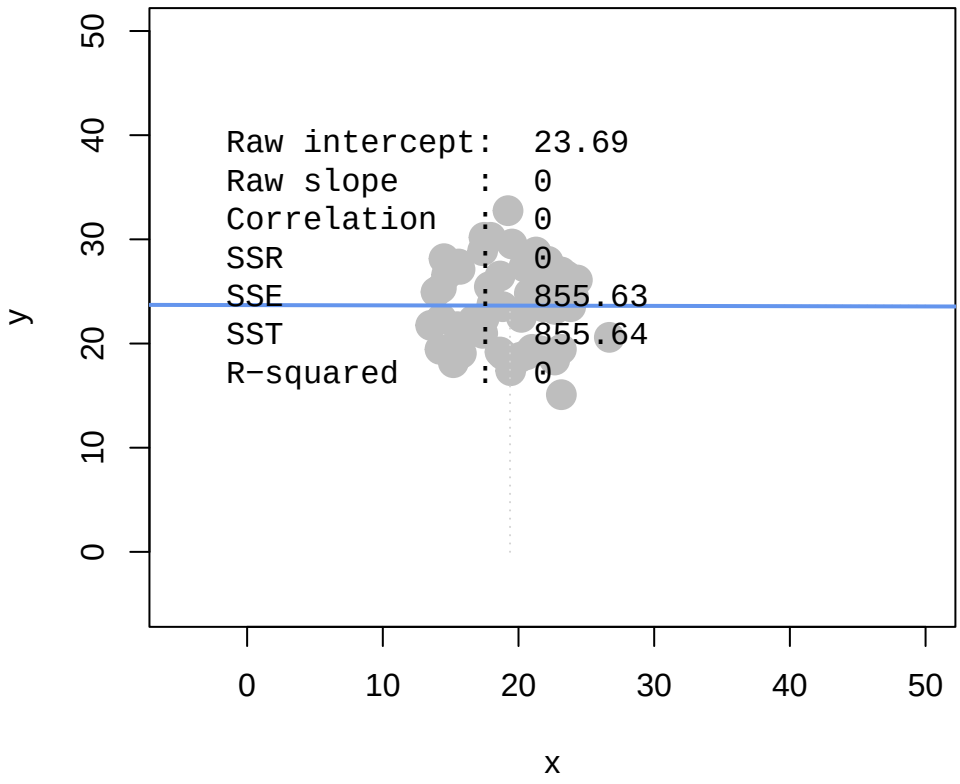
A: -1

ii) What is the correlation of x and y that you would generally expect?

A: -1

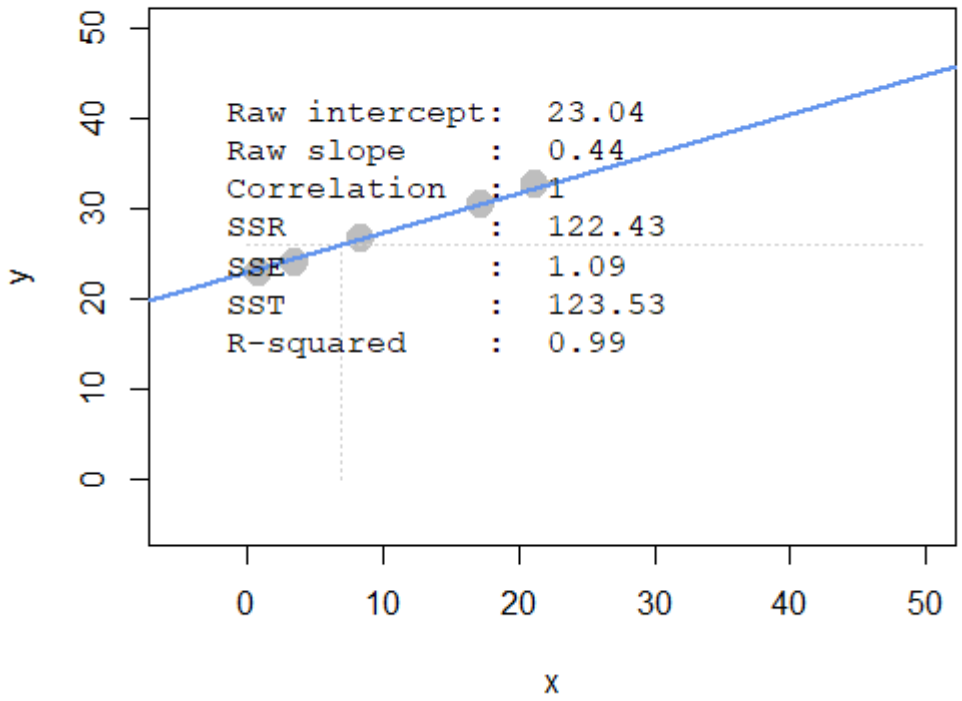
e) Apart from any of the above scenarios, find another pattern of data points with no correlation (r 0).

This is a pattern of a circle.

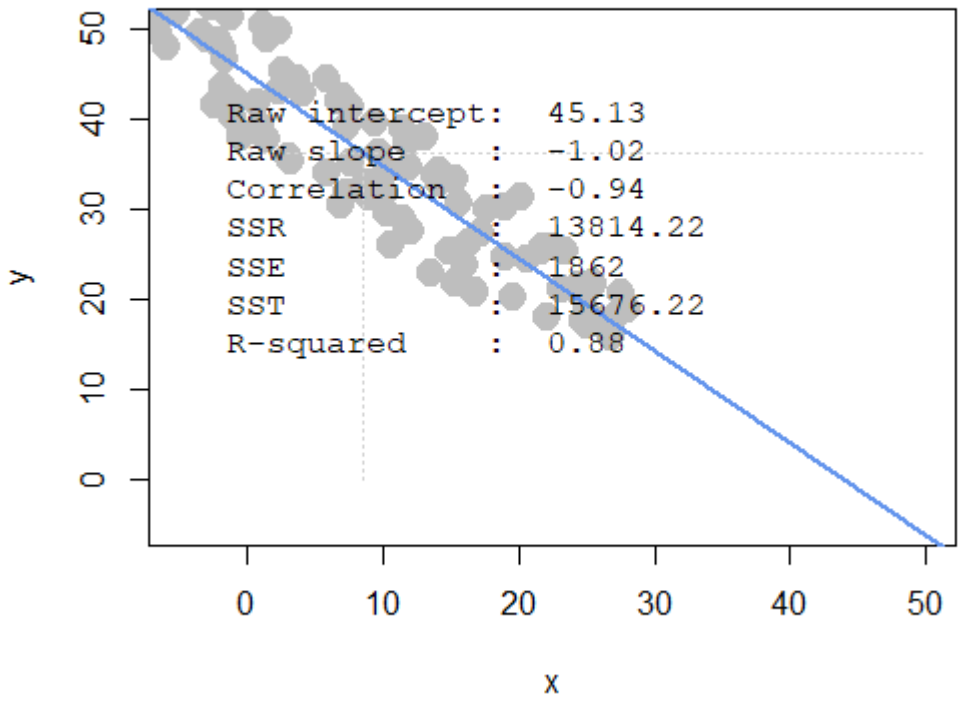


f) Apart from any of the above scenarios, find another pattern of data points with perfect correlation (r 1).

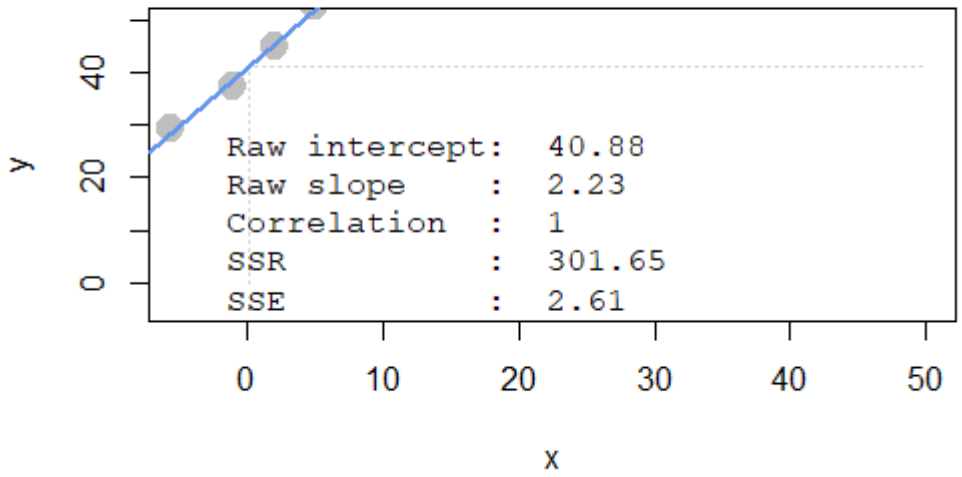
This is a pattern of a straight line.


Raw intercept: 23.04
Raw slope : 0.44
Correlation : 1
SSR : 122.43
SSE : 1.09
SST : 123.53
R-squared : 0.99
y
x
0
10
20
30
40
50

g) Let's see how correlation relates to simple regression, by simulating any linear relationship you wish:



i) Run the simulation and record the points you create: pts <- interactive_regression() (simulate e



ii) Use the lm() function to estimate the regression intercept and slope of pts to ensure they are

```
Call:
lm(formula = pts$y ~ pts$x)

Residuals:
      1       2       3       4
 0.7605 -1.1733 -0.3266  0.7395

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  40.8793     0.5713   71.56 0.000195 ***
pts$x         2.2321     0.1468   15.21 0.004296 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.142 on 2 degrees of freedom
Multiple R-squared:  0.9914,    Adjusted R-squared:  0.9871
F-statistic: 231.3 on 1 and 2 DF,  p-value: 0.004296
```

iii) Estimate the correlation of x and y to see it is the same as reported in the plot: cor(pts)

```
  x        y
```

x 1.000000 0.995704 y 0.995704 1.000000

The result is close to 1, which is same as reported in the plot.

iv) Now, standardize the values of both x and y from pts and re-estimate the regression slope

```
Call:
lm(formula = pts$y ~ pts$x)

Residuals:
      1       2       3       4
 0.7605 -1.1733 -0.3266  0.7395

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  40.8793     0.5713   71.56 0.000195 ***
pts$x         2.2321     0.1468   15.21 0.004296 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.142 on 2 degrees of freedom
Multiple R-squared:  0.9914,    Adjusted R-squared:  0.9871
F-statistic: 231.3 on 1 and 2 DF,  p-value: 0.004296
```

v)What is the relationship between correlation and the standardized simple-regression estimates?

The standardlized estimates value for x equals to the correlation.