# Exploring Prerequisite Relationships between Mathematical Concepts in Intelligent Tutoring Systems

**Elaine Xu, Smeet Poladia, Zhou Lu**

## Abstract

Intelligent Tutoring Systems (ITSs) are an educational technology that provides students with a virtual learning environment, where every action of a student is tracked and recorded as a transaction. In Mathia, an ITS for school mathematics developed by Carnegie Learning, Students ~~deal with math~~ are presented with workspaces , groups of problems on the same topic. ~~in these tutoring systems, which have problems, and~~ Each problem further has many steps. Whenever a student solves a step correctly, they may learn a Knowledge Component (KC). ~~and~~ In this paper, we investigate prerequisite relationships between KCs. We build ~~observe~~ Gaussian Graphical Models (GGM) to identify strongly related KCs ~~and decide the metric of student learning based on the best structure we observe in the GGM.~~ To determine prerequisite relationships, we use mixed-effects logistic regression and inspect the statistical significance of the coefficient of the main effect. Based on this approach, we know the prerequisite relation between KCs among student data on three workspaces.

## 1. Introduction

Intelligent Tutoring Systems (ITS) are an educational technology that provides students with a virtual learning environment and logs data of students' learning experience. Every action a student makes - from answering a question to requesting hints - ~~would be~~ is tracked by the system and recorded as a transaction. Add some background here about what Mathia is, what Carnegie Learning is, who your client is, etc. The project aims at making ITS more effective, by detecting ~~and the core problem is to determine~~ whether prerequisite relations among math topics can be detected in log data. If we can determine that mastering a certain math skill is necessary or can facilitate the learning of another skill, then the tutor system could train students on this prerequisite skill first and yield a better learning experience.

Some specific questions to be addressed in the paper are:

- How do we determine whether two math skills are related?
- What metrics of learning and performance? Not a sentence.
- How do we test whether topic/skill/unit A is prerequisite for B?

make sure you return to these questions (state each one explicitly and show how what you are doing helps to answer that question) in each section of the paper, especially methods, results and discussion.

## 2. Data

The data for this study are provided by Carnegie Learning. Four datasets are used in this study. The first three datasets provide information about a random sample of students' performance on three different workspaces in 2019-2020 academic year. The three workspaces are "A:

Analyzing Models of Two-Step Linear Relationships”, “B: Modeling Two-Step Expressions”, and “C: Using Scale Factor”. In this study, we assume that the knowledge components in workspace “A: Analyzing Models of Two-Step Linear Relationships” are prerequisite for knowledge components in workspace “B: Modeling Two-Step Expressions”. Moreover, we believe that the “C: Using Scale Factor” workspace should not be a prerequisite of two workspaces A and B, and comes prior to them in the curriculum. The reader should refer to Corbett et al., (2000) for details about this prerequisite relationship. The fourth dataset is a larger dataset which contains a random sample of 500 students’ performance in all Course 2 (Grade 7) MATHia workspaces in the 2019-2020 academic year.

need parens, no comma

| Dataset | # of Students | # of Unique Knowledge Components | # of Unique Steps (Opportunities) |
|---|---|---|---|
| A: Analyzing Models of Two-Step Linear Relationships | 29949 | 7 | 7 |
| B: Modeling Two-Step Expressions | 27005 | 9 | 9 |
| C: Using Scale Factor | 19521 | 4 | 29 |
| MATHia Course 2 Sample | 500 | 964 | 117210 |

close up space

Table 1: General Overview of Datasets

Table 1 provides some general information about the number of unique students, the number of unique knowledge components, and the number of unique steps (opportunities) in each data set.

Put text referring to table before table, if at all possible

| Variable Name | Values | Description |
|---|---|---|
| Anno.Student.Id | Integer | anonymous student identifier |
| Time | Timestamp | Timestamp in UNIX epoch time |
| Problem.Name | Character | Identifier for the problem |
| Step.Name | Character | Identifier for the problem-step |
| Action | “Attempt”,”Done”, “Hint | Student’s action for the problem-step. “Attempt” = student made a problem-solving attempt, “Done” = |

|  |  |  |
|---|---|---|
|  | Request", "Hint Level Change" | student clicked the "Done" button required to complete a problem, "Hint Request" = Student requests a hint, "Hint Level Change" = Student requests a hint at a "deeper" level |
| Outcome | "OK",<br>"ERROR",<br>"BUG",<br>"INITIAL_HINT",<br>"HINT_LEVEL_CHANGE" | "OK" = correct, "ERROR" = error that isn't specifically tracked for JIT feedback, "BUG" = error that is tracked for just-in-time, context-sensitive feedback, "INITIAL_HINT" = first-level hint is provided, "HINT_LEVEL_CHANGE" = a "deeper" level hint is provided |
| KC..MATHia. | Character | The skill or knowledge component (KC) tracked by MATHia for this problem-step |
| CF (Skill New p-Known) | Real Number | BKT skill estimate after this action (i.e., semantic event) |

Table 2: Descriptions of Variables in datasets

ASA style please!

In Table 2, we provide descriptions of selected variables. Within each dataset, there are multiple problems. Within each problem, multiple steps (opportunities) are presented to students. A unique knowledge component is mapped to each step. Readers can refer to Fancsali et al., 2021 for more details about the variable descriptions.

Did you forget workspaces here?

# 3. Methods

## Gaussian Graphical Models

cite the paper on ggm's here

FIML is only one method; you are using others too.

What outcome measure are you using for KC's? Etc.

A gaussian graphical model (GGM) is an exploratory analysis tool that provides an easy to grasp overview of relationships between knowledge components (KCs) from workspaces in an intelligent tutoring system. The GGM uses correlations between KCs calculated using the Full Information Maximum Likelihood (FIML) criteria. Using this correlation matrix as an input, the graphical model comprises KCs depicted by circles and a set of lines that visualize the relationship between the KCs. The thickness of these lines represents the strength of relationships and the correlations can be interpreted as partial correlation coefficients. The line has green color if the correlation is positive and red otherwise. For example, we see in Figure 1, a thick green line between KC 1 and KC 5. This implies a strong association between these two

Need to talk about partial correlations

KCs, after eliminating for the effect of other KCs. To further learn about the prerequisite structure between these KCs, we use mixed-effects logistic regression models. GGM assumes an underlying multivariate normal distribution for the data, and further details are mentioned in the Discussion section of this paper.

Doesn't belong here

### Initial opportunities

To better understand students' performance on each knowledge component, for each student, we used initial opportunities instead of all opportunities to evaluate that student's understanding of mathematical concepts. For each knowledge component, students are given multiple opportunities (steps) until students demonstrate mastery of that mathematical skill. Different numbers of opportunities are given to each student based on their performance. In order to prevent smoothing out differences among students, we chose to use initial opportunities which are better indications of students' mastery level.

We utilized Gaussian Graphical Models to determine the cutoff point for initial opportunities. We generated multiple Gaussian Graphical Models using different cutoff points. The number of opportunities which produces the model with the best structure is selected as our final cutoff point for initial opportunities.

Say what criterion you used to determine "best structure" here

### Mixed Effects Logistic Regression

To better understand the partial correlations between two KCs from the Gaussian Graphical Model and to quantify the influence one KC has on another, we adopted a mixed effects logistic regression approach. If there exist some prerequisite relationships between two knowledge components KC1 and KC2, and we assume KC2 is a prerequisite of KC1, then student who knows KC2 should have better performance on KC1 than student who does not know KC2. At the same time, whether a student knows KC1 should not have much effect on student's performance on KC2.

I thought you were using indicator of whether the student had mastered KC1?

We used First Attempt of KC 1 as the dependent variable, which is a binary variable that indicates whether a student answered the step correctly on their first attempt. Predictors include KC 1 opportunity - the number of times a student has encountered KC 1, whether a student has mastered KC 2, and an interaction term. Student ID is the random effect term since its variability cannot be explained by the predictors of the model. The tutor system calculates a score of 0 to 1 to indicate a student's grasp of a certain KC (CF.Skill.New.P.Known), and deems a student has mastered this KC if the score is above 0.95. Since the opportunity column only exists in a side dataset which includes 500 students, we used this smaller dataset instead of the complete ones mentioned in the data section.

This is too late to mention a new data set. This should be in the data section.

The final model looks like:

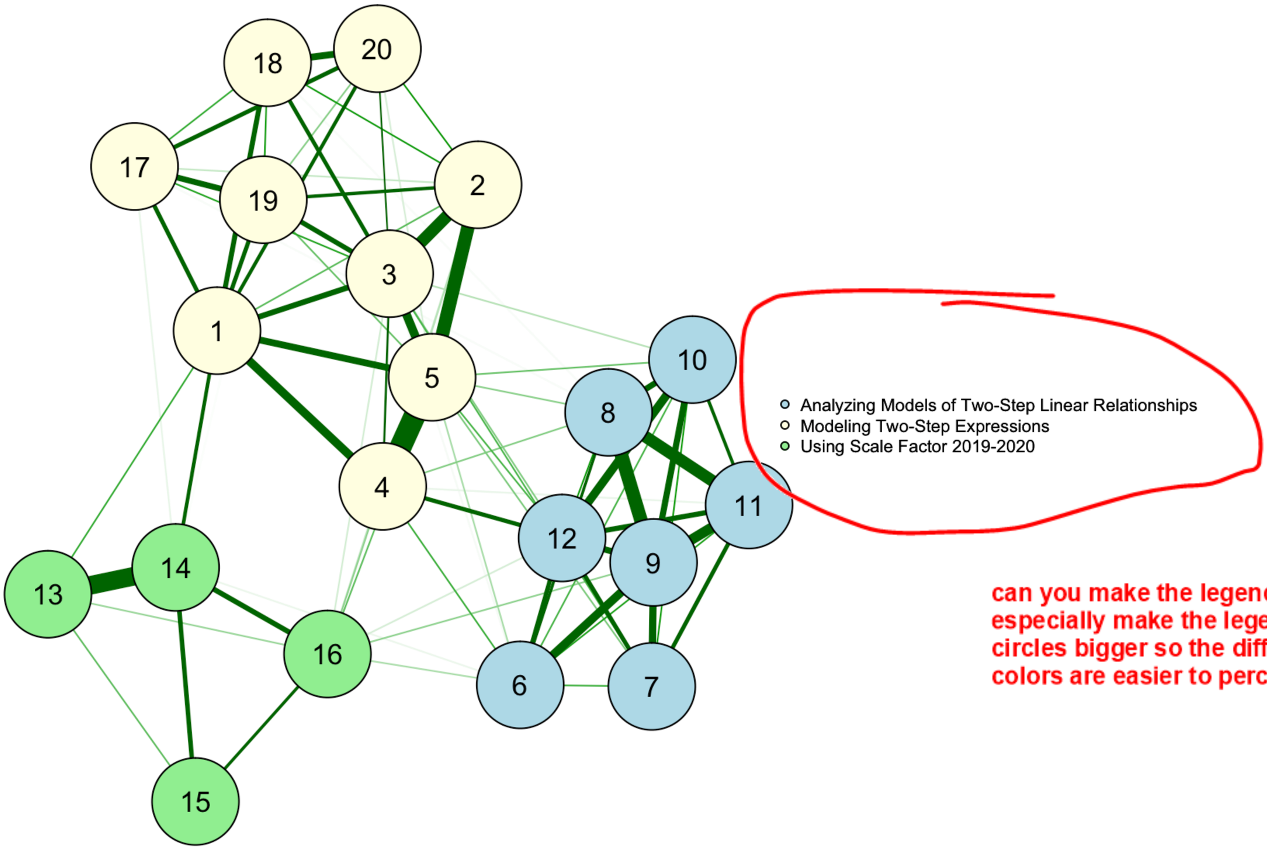
*glmer(First Attempt of KC1 ~ 1 + KC1 Opportunity + know_KC2 + KC1 Opportunity : know_KC2 + (1|Anon.Student.Id))*

## 4. Results

### Gaussian Graphical Models

The following graphical model (Figure 1) is of success rates of students on initial 2 opportunities for KCs on the three workspaces: A: Analyzing Models of Two-Step Linear Relationships", "B: Modeling Two-Step Expressions", and "C: Using Scale Factor".

do you transform this success rate?


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
● Analyzing Models of Two-Step Linear Relationships
○ Modeling Two-Step Expressions
● Using Scale Factor 2019-2020

can you make the legend bigger - especially make the legend circles bigger so the different colors are easier to percieve.

Figure 1: Gaussian Graphical Model of Success Rates on Initial 2 Opportunities

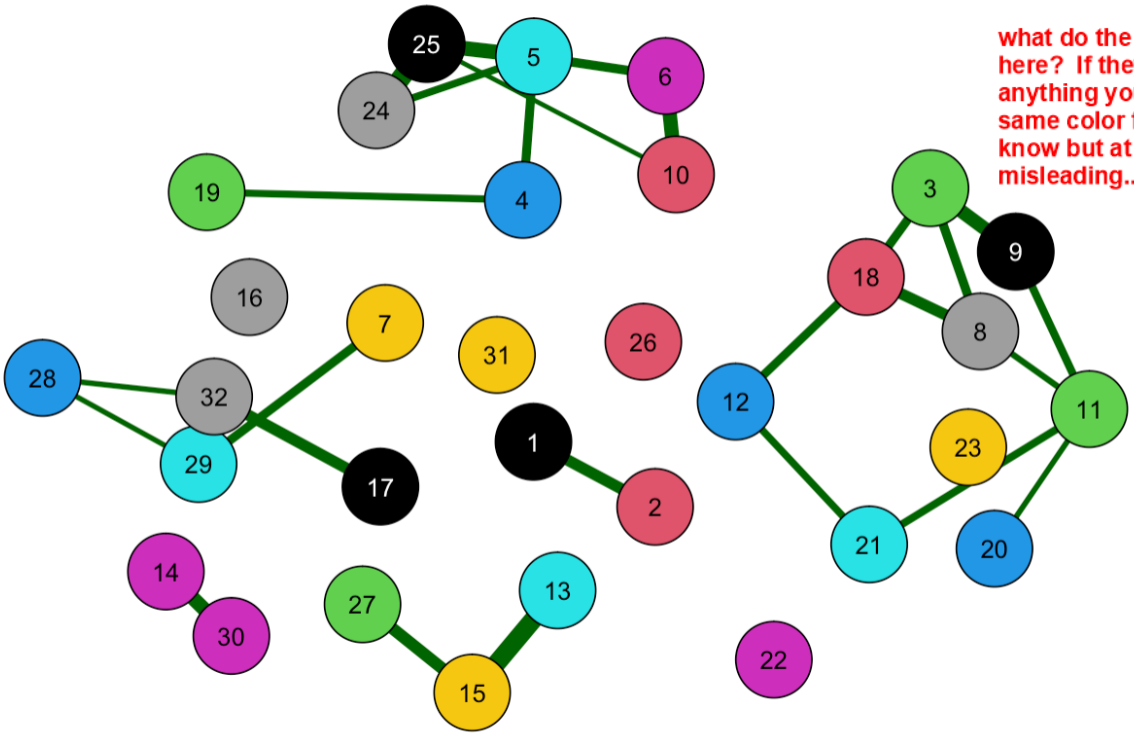
| KC Number | KC | Workspace |
|---|---|---|
| | | |
| 1 | define variable-1 | B |
| 2 | enter given, reading numerals-1 | B |
| 3 | enter given, reading words-1 | B |
| 4 | find y, any form-1 | B |
| 5 | identifying units-1 | B |
| 6 | interpret scenario with numbers | A |
| 7 | interpret scenario with words | A |
| 8 | match _dep expression with description. | A |
| 9 | match _indep expression with description. | A |
| 10 | match _intercept expression with description. | A |
| 11 | match _linear-term expression with description. | A |
| 12 | match _slope expression with description. | A |
| 13 | scale-drawings-3-determine unknown measure, complex scale factor. | C |
| 14 | scale-drawings-3-determine unknown measure, simple scale factor. | C |
| 15 | scale-drawings-3-enter scale factor units. | C |
| 16 | scale-drawings-3-enter scale factor value. | C |
| 17 | write expression, negative intercept-1 | B |
| 18 | write expression, negative slope-1 | B |
| 19 | write expression, positive intercept-1 | B |
| 20 | write expression, positive slope-1 | B |

what do the "-1" at the end of some names mean?

why do some have underscores, and others do not?

Table 3: Description of table needed

The following graphical model (Figure 2) is of success rates of students on initial 2 opportunities of intentionally chosen KCs from the Course 2 (Grade 7) MATHia workspaces in the 2019-2020 academic year. The KCs which had similar mathematical meaning were chosen, such that we would expect prerequisite relationships among them. Also, some KCs were added randomly which should not be a prerequisite of any other KCs. which ones?



what do the colors mean here? If they don't mean anything you should use the same color for all (boring I know but at least not misleading...)

Figure 2: description of figure needed

| KC Number | KC |
|---|---|
| | |
| 1 | select combine like terms-1 |
| 2 | combine like terms-1 |
| 3 | select perform division-1 |
| 4 | perform division |
| 5 | select simplify fractions-1 |

| | |
|---|---|
| 6 | select apply exponent-1 |
| 7 | apply exponent-1 |
| 8 | select perform multiplication-1 |
| 9 | perform multiplication-1 |
| 10 | select combine like terms, unary coefficient |
| 11 | select combine like terms within parens |
| 12 | enter given triangular prism dimension of base |
| 13 | enter given prism volume |
| 14 | find area of base of triangular prism |
| 15 | find prism height |
| 16 | work with triangular prism in standard position |
| 17 | enter given prism height |
| 18 | enter given rectangular prism dimension of base |
| 19 | find rectangular prism dimension of base |
| 20 | find area of base of rectangular prism |
| 21 | work with prism in context |
| 22 | find triangular prism dimension of base |
| 23 | work with prism out of context |
| 24 | enter given pyramid side length of base |
| 25 | enter given pyramid height |

| | |
|---|---|
| 26 | find area of base of pyramid |
| 27 | find pyramid volume |
| 28 | work with pyramid in standard position |
| 29 | enter given pyramid volume |
| 30 | find pyramid height |
| 31 | work with pyramid out of context |
| 32 | work with pyramid in context |

Table 4: Mathematical Interpretations for Knowledge Components

what do the "-1" at the end of some names mean?

**Initial Opportunities**

After comparison, the first two initial opportunities gave us the best structured Gaussian Graphical Model (Figure 1). Therefore, for the rest of the analysis, we are going to use the first two initial opportunities to evaluate students' performance. More details about the Gaussian Graphical Models for different numbers of opportunities are available in Appendix 2.

please give page numbers in the appx

**Mixed Effects Logistic Regression**

For simplicity, the glmer method was first applied on data of workspace B: "Modeling Two-Step Expressions". Glmer was applied to different pairs of knowledge components, and the model results were recorded in Table 5. Main effect indicates if knowing kc 2 would influence student's performance on kc 1, and the interaction term tells us if knowing kc 2 would make learning kc 1 faster. The coefficients in red are not statistically significant (p-value > 0.05).

The results align with what we saw in the Gaussian Graphical Models. For example, there is no line connecting KC1 and KC18 in Figure 3, and the coefficients of main effect in Table 5 for them are mostly not significant. It is also interesting to note that the coefficients for interaction terms are all negative, which means learning a certain KC would have a negative effect on how fast students learn another KC.

| "Prereq" KC | KC | Main Effect | Opportunity | Interaction |
|---|---|---|---|---|
| 1 | 5 | 1.21 | 0.18 | -0.13 |
| 5 | 1 | 2.43 | 0.34 | -0.31 |
| 4 | 5 | 1.20 | 0.10 | -0.07 |
| 5 | 4 | 0.50 | 0.08 | -0.04 |
| 18 | 1 | 1.45 | 0.63 | -0.60 |
| 1 | 18 | 30.98 | 28.63 | -28.90 |
| 18 | 5 | 1.20 | 0.12 | -0.08 |
| 5 | 18 | 4.00 | 1.99 | -1.64 |

Table 5: Glmer Results

Much more explanation and discussion of this table (or one like it) will be needed in the final paper. A person can't really tell what the column names or the entries in the table mean with what you have explained in this draft paper.

## 5. Discussion

A limitation of GGM for this project is that the method assumes an underlying multivariate normal distribution for the data. We applied logit transformation to transform the distribution of variables (success rate for initial opportunities for each KC). For some KCs, the transformed data points were normally distributed, but for other KCs, we observed a distribution that had a lot of data points at the end tails. This is because of observations having actual success rates of either 0 or 1. Before applying logit transformation, all 0s were changed to 0.0001, and all 1s were changed to 0.9999.

Even though the glmer method is a good way to quantify correlations between knowledge components, it does not infer causal relationships. Time order is a challenge here since it does not take into account the order of a student learning certain knowledge components. The order of topics is usually fixed, which makes it harder to test for prerequisites.

## 6. References

replace "et al" with full author list, on all papers. You can use "et al" in the main text when you refer to this article

format in ASA style please!

Bhushan, Nitin et al. 2019. "Using a Gaussian Graphical Model to Explore Relationships Between Items and Variables in Environmental Psychology Research." Frontiers in Psychology 10.

Gauthier, Gilles, Claude Frasson, and Kurt VanLehn. 2000. Intelligent Tutoring Systems: 5th International Conference, ITS 2000, Montreal, Canada, June 19-23, 2000: Proceedings. Berlin: Springer.

Stephen et al. 2021. “Carnegie Learning MATHia 2019-2020 DataShop Documentation.”

Koedinger, K. R., Baker, R. S. J. d., Cunningham, K., Skogsholm, A., Leber, B., & Stamper, J. (2010). A data repository for the EDM community: The PSLC datashop. In S. Ventura, C. Romero, M. Pechenizkiy, & R. S. J. d. Baker (Eds.), Handbook of educational data mining (pp. 43–55). Boca Raton, FL: CRC Press.

R Core Team (2013). R: A language and environment for statistical computing.

# Appendices

## Content

This is a good start, but you will need to add English to help reader understand what you did and why (and what the result was)

## Appendix 1. Gaussian Graphical Models

Data import and data wrangling

```
mathia <- read.csv(file = "HCI_Prerequisite_Relations/data_code/data/MATHia_Course_2_(Grade-7)-All_Data_2019-2020/course2_1920_sample1_500students_datashop.txt", header = TRUE, sep = "\t")

kcs <- unique(mathia$KC.Model.MATHia.)[c(46:56,123:143)]

kcs

##  [1] "select combine like terms-1"

##  [2] "combine like terms-1"

##  [3] "select perform division-1"

##  [4] "perform division"

##  [5] "select simplify fractions-1"

##  [6] "select apply exponent-1"

##  [7] "apply exponent-1"

##  [8] "select perform multiplication-1"

##  [9] "perform multiplication-1"

## [10] "select combine like terms, unary coefficient"

## [11] "select combine like terms within parens"

## [12] "enter given triangular prism dimension of base"

## [13] "enter given prism volume"

## [14] "find area of base of triangular prism"

## [15] "find prism height"
## [16] "work with triangular prism in standard position"
## [17] "enter given prism height"
## [18] "enter given rectangular prism dimension of base"
## [19] "find rectangular prism dimension of base"
## [20] "find area of base of rectangular prism"
## [21] "work with prism in context"
## [22] "find triangular prism dimension of base"
## [23] "work with prism out of context"
## [24] "enter given pyramid side length of base"
## [25] "enter given pyramid height"
## [26] "find area of base of pyramid"
## [27] "find pyramid volume"
## [28] "work with pyramid in standard position"
## [29] "enter given pyramid volume"
## [30] "find pyramid height"
## [31] "work with pyramid out of context"
## [32] "work with pyramid in context"
new_mathia <- mathia %>%
  filter(KC.Model.MATHia. %in% kcs)
new_mathia <- new_mathia %>%
  select(Anon.Student.Id, KC.Model.MATHia., Outcome) %>%
  group_by(Anon.Student.Id)
new_mathia <- new_mathia %>%
  pivot_wider(names_from = KC.Model.MATHia., values_from = Outcome)
revised_mathia <- mathia %>%
  filter(KC.Model.MATHia. %in% kcs) %>%
  select(Anon.Student.Id, KC.Model.MATHia., Step.Name, Level..Workspace.Id.) %>%
  count(Anon.Student.Id, KC.Model.MATHia., Step.Name, Level..Workspace.Id.)
revised_mathia <- revised_mathia %>%
  pivot_wider(names_from = KC.Model.MATHia., values_from = n)
new_revised_mathia <- revised_mathia %>%
  select(-Step.Name, -Level..Workspace.Id.) %>%
  group_by(Anon.Student.Id) %>%
  mutate("select combine like terms-1" = list(`select combine like terms-1`),
         "combine like terms-1" = list(`combine like terms-1`),
         "select perform division-1" = list(`select perform division-1`),
         "perform division" = list(`perform division`),
         "select simplify fractions-1" = list(`select simplify fractions-1`),
         "select apply exponent-1" = list(`select apply exponent-1`),
         "apply exponent-1" = list(`apply exponent-1`),


         "select perform multiplication-1" = list(`select perform multiplication-1`),
         "perform multiplication-1" = list(`perform multiplication-1`),
         "select combine like terms, unary coefficient" = list(`select combine like terms, unary coefficient`),
         "select combine like terms within parens" = list(`select combine like terms within parens`),
         "enter given triangular prism dimension of base" = list(`enter given triangular prism dimension of base`),
         "enter given prism volume" = list(`enter given prism volume`),
         "find area of base of triangular prism" = list(`find area of base of triangular prism`),


         "find prism height" = list(`find prism height`),
         "work with triangular prism in standard position" = list(`work with triangular prism in standard position`),
         "enter given prism height" = list(`enter given prism height`),
         "enter given rectangular prism dimension of base" = list(`enter given rectangular prism dimension of base`),
         "find rectangular prism dimension of base" = list(`find rectangular prism dimension of base`),
         "find area of base of rectangular prism" = list(`find area of base of rectangular prism`),
         "work with prism in context" = list(`work with prism in context`),

         "find triangular prism dimension of base" = list(`find triangular prism dimension of base`),
         "work with prism out of context" = list(`work with prism out of context`),
         "enter given pyramid side length of base" = list(`enter given pyramid side length of base`),
         "enter given pyramid height" = list(`enter given pyramid height`),
         "find area of base of pyramid" = list(`find area of base of pyramid`),
         "find pyramid volume" = list(`find pyramid volume`),
         "work with pyramid in standard position" = list(`work with pyramid in standard position`),

         "enter given pyramid volume" = list(`enter given pyramid volume`),
         "find pyramid height" = list(`find pyramid height`),
         "work with pyramid out of context" = list(`work with pyramid out of context`),
         "work with pyramid in context" = list(`work with pyramid in context`)) %>%
  distinct()
opp_mathia <- new_revised_mathia
remove_na <- function(v){
  v1 <- v[!is.na(v)]
  if (is_empty(v1) == TRUE){
    return(0)
  }
  return(v1)
```

```r
}
for (i in colnames(opp_mathia)[-1]){
  opp_mathia[[i]] <- lapply(opp_mathia[[i]], remove_na)
}
get_opp_success_rate <- function(v1, v2){
  if (v1 == 0 | is.null(v2)==TRUE){
    return(0)
  }
  success <- function(vec){
    ret_vec <- length(which(vec %in% "OK"))/length(vec)
    return(ret_vec)
  }
  ret_fin_vec <- c()
  v <- v2
  for (i in 1:length(v1)){
    ret_fin_vec <- c(ret_fin_vec, success(v[1:v1[i]]))
    v <- v[-c(1:v1[i])]
  }
  return(ret_fin_vec)
}
full_mathia <- cbind(opp_mathia, new_mathia)
opp_opp_mathia <- list()
for (j in 2:33){
  temp <- full_mathia[,c(j,j+33)]
  x <- c(temp[[1]], temp[[2]])
  temp_res <- list()
  for (i in 1:489){
    temp_res[[i]] <- get_opp_success_rate(x[[i]],x[[i+489]])
  }
  opp_opp_mathia[[j-1]] <- temp_res
}
tp_opp <- opp_mathia
for (i in 1:32){
  tp_opp[[i+1]] <- as.vector(opp_opp_mathia[[i]])
}
```

Loading packages for GGM

```r
require(tidyverse)
require(qgraph)
require(xtable)
require(dplyr)
require(bootnet)
require(rstudioapi)
```

Logit and GGM function

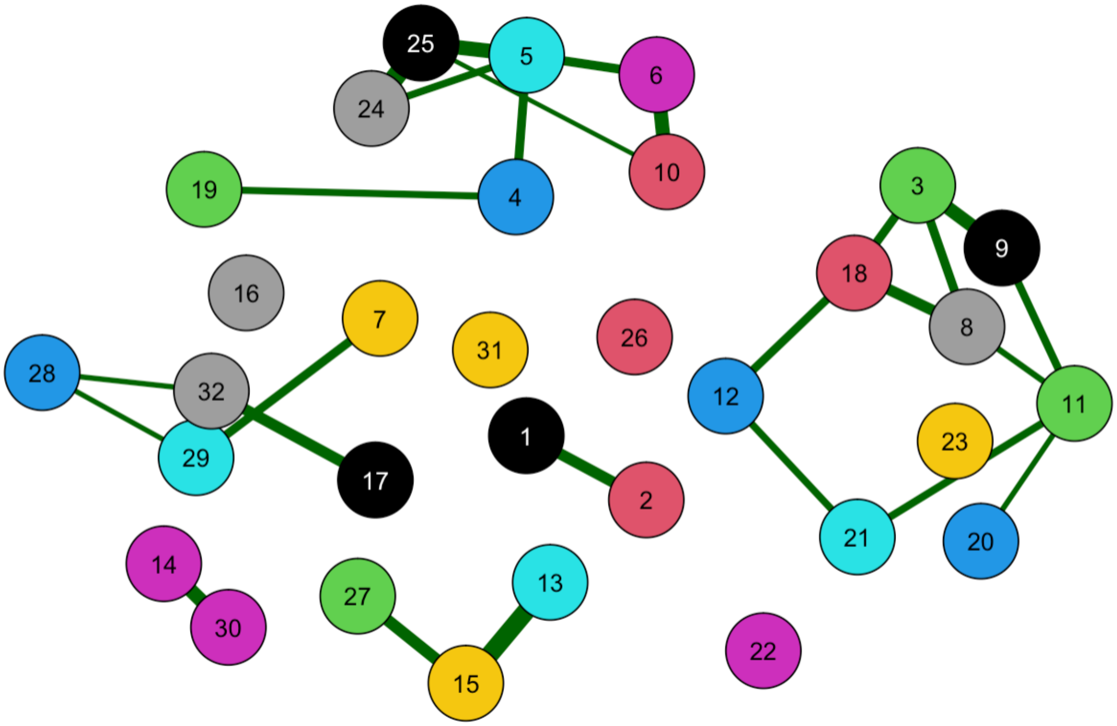
```r
logit <- function(v){
  if (is.na(v)==TRUE){
    return(NA)
  }
  if (is.nan(v) == TRUE){
    return(NA)
  }
  if (v==1){
```

```
  v = 0.9999

 }

 if (v==0){

  v = 0.0001

 }

 return(log(v/(1-v)))

}

ggm_opp <- tp_opp[,-1]

ggm <- function(dat, num, min_cor = 0.03){
 success_initial_opp <- function(v){
   if(length(v) >= num){
    v1 <- as.numeric(v[1:num])
    success <- sum(v1)/num
    return(success)
   }
   else {v1 <- as.numeric(v)}
   success <- sum(v1)/length(v1)
   return(success)
  }
 opp_data_success_initial <- dat
 for (i in colnames(opp_data_success_initial)){
  opp_data_success_initial[[i]] <- lapply(opp_data_success_initial[[i]], success_initial_opp)
 }
 new_opp_data_success_initial <- opp_data_success_initial
 for (i in colnames(new_opp_data_success_initial)){
   new_opp_data_success_initial[[i]] <- lapply(new_opp_data_success_initial[[i]], logit)
 }
 for (i in 1:ncol(new_opp_data_success_initial)){
   new_opp_data_success_initial[[i]] <- as.numeric(new_opp_data_success_initial[[i]])
 }
 new_opp_data_success_initial <- na.omit(new_opp_data_success_initial)
 data_success_initial_corr <- psych::corFiml(new_opp_data_success_initial)
 qgraph::qgraph(
   data_success_initial_corr,
   layout = "spring",
   graph = "glasso",
   labels = TRUE,
   legend.cex = 0.30,
   tuning = 0.1,
   color = c(1:32),
   labels = TRUE,
   sampleSize = 489,
   minimum = min_cor
  )
}
```

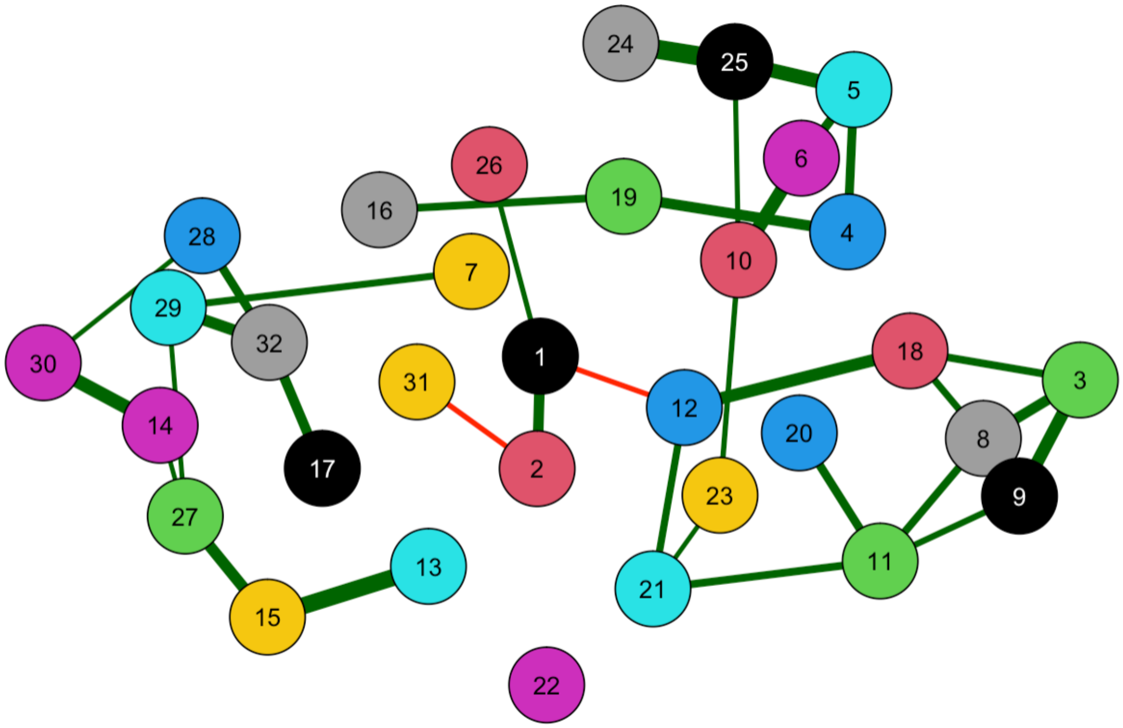
Gaussian Graphical Models

In the following plots, for the sake of easier visual comparison, we only show partial correlations that are larger than 0.15.

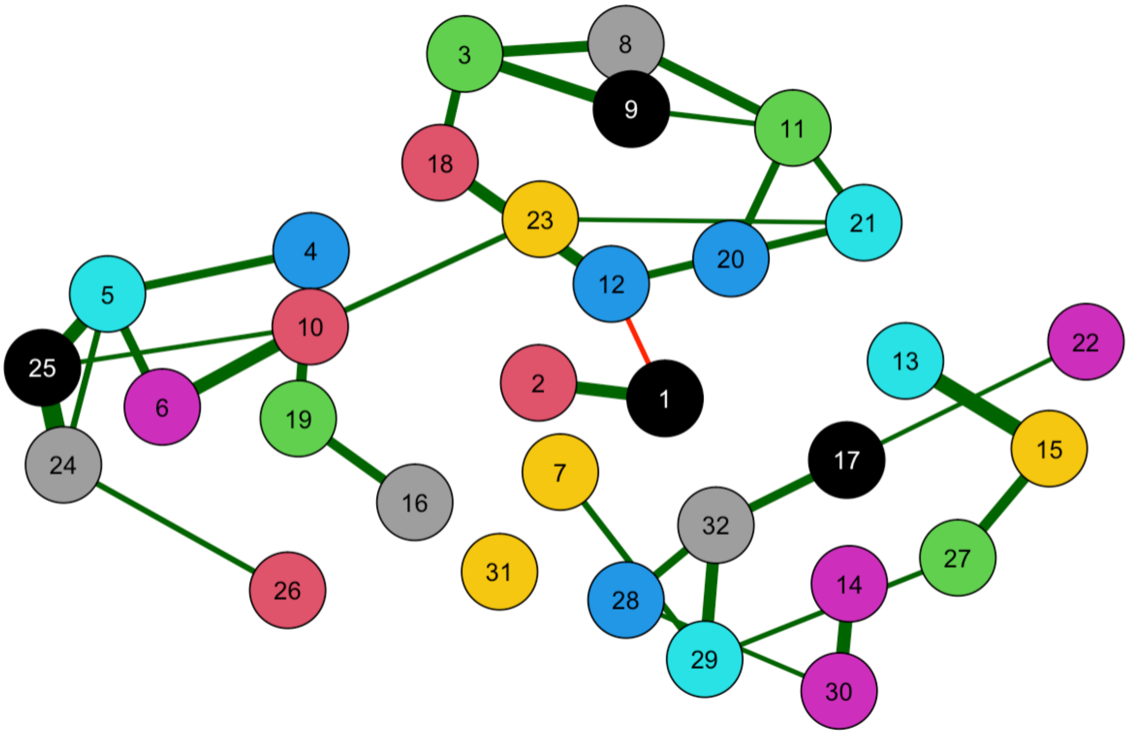
```
ggm(dat = ggm_opp,num = 2, min_cor = 0.15)
```



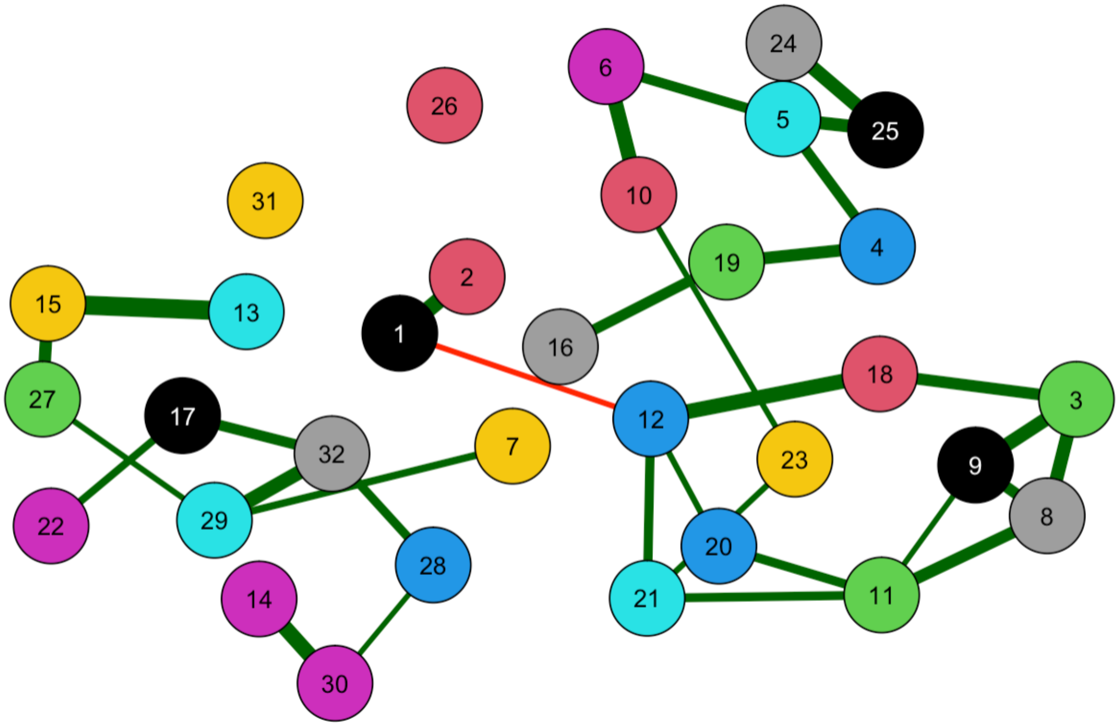
```
ggm(dat = ggm_opp,num = 3, min_cor = 0.15)
```



```
ggm(dat = ggm_opp,num = 4, min_cor = 0.15)
```



```
ggm(dat = ggm_opp,num = 5, min_cor = 0.15)
```



## Appendix 2. Initial Opportunities

Read data and load required packages.

```
require(tidyverse)

require(qgraph)

require(xtable)

require(dplyr)

require(bootnet)

require(rstudioapi)

load("ggm_function.RData")

ggm_opp <- tp_opp[-1]
```

Apply logit transformation to transform the success rate for initial opportunities for each KC. All 0s were changed to 0.0001, and all 1s were changed to 0.9999.

```
logit <- function(v){
 if (is.na(v)==TRUE){
        return(NA)
 }
 if (is.nan(v) == TRUE){
        return(NA)
 }
 if (v==1){
        v = 0.9999
 }
 if (v==0){
        v = 0.0001
 }
 return(log(v/(1-v)))
}
```

Use Gaussian Graphical Model to generate the plot.

```
ggm <- function(dat, num, min_cor){
 success_initial_opp <- function(v){
        if(length(v) >= num){
        v1 <- as.numeric(v[1:num])
        success <- sum(v1)/num
        return(success)
        }
        else {v1 <- as.numeric(v)}
        success <- sum(v1)/length(v1)
```

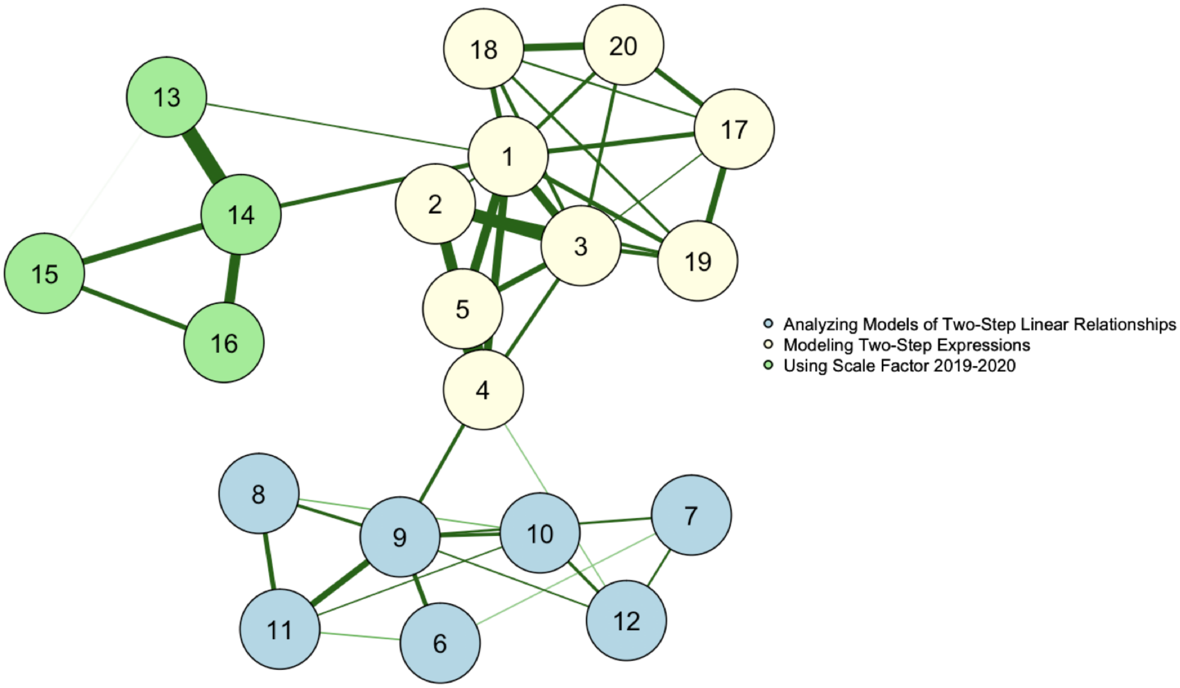
```r
    return(success)
    }
opp_data_success_initial <- dat
for (i in colnames(opp_data_success_initial)){
    opp_data_success_initial[[i]] <- lapply(opp_data_success_initial[[i]], success_initial_opp)
}
new_opp_data_success_initial <- opp_data_success_initial
for (i in colnames(new_opp_data_success_initial)){
    new_opp_data_success_initial[[i]] <- lapply(new_opp_data_success_initial[[i]], logit)
}
for (i in 1:ncol(new_opp_data_success_initial)){
    new_opp_data_success_initial[[i]] <- as.numeric(new_opp_data_success_initial[[i]])
}
new_opp_data_success_initial <- na.omit(new_opp_data_success_initial)
data_success_initial_corr <- psych::corFiml(new_opp_data_success_initial)
group_items <- list(
    `Analyzing Models of Two-Step Linear Relationships` = c(6:12),
    `Modeling Two-Step Expressions` = c(1:5,17:20),
    `Using Scale Factor 2019-2020` = c(13:16)
    )
qgraph::qgraph(
    data_success_initial_corr,
    layout = "spring",
    graph = "glasso",
    labels = TRUE,
    legend.cex = 0.32,
        tuning = 0.1,

        color = c("light blue", "light yellow", "light green"),

        groups = group_items,

        labels = TRUE,

        sampleSize = 10761,

        minimum = min_cor

        )

}
```

Generate Gaussian Graphical Models for different numbers of initial opportunities:

In the following plots, for the sake of easier visual comparison, we only show partial correlations that are larger than 0.05.

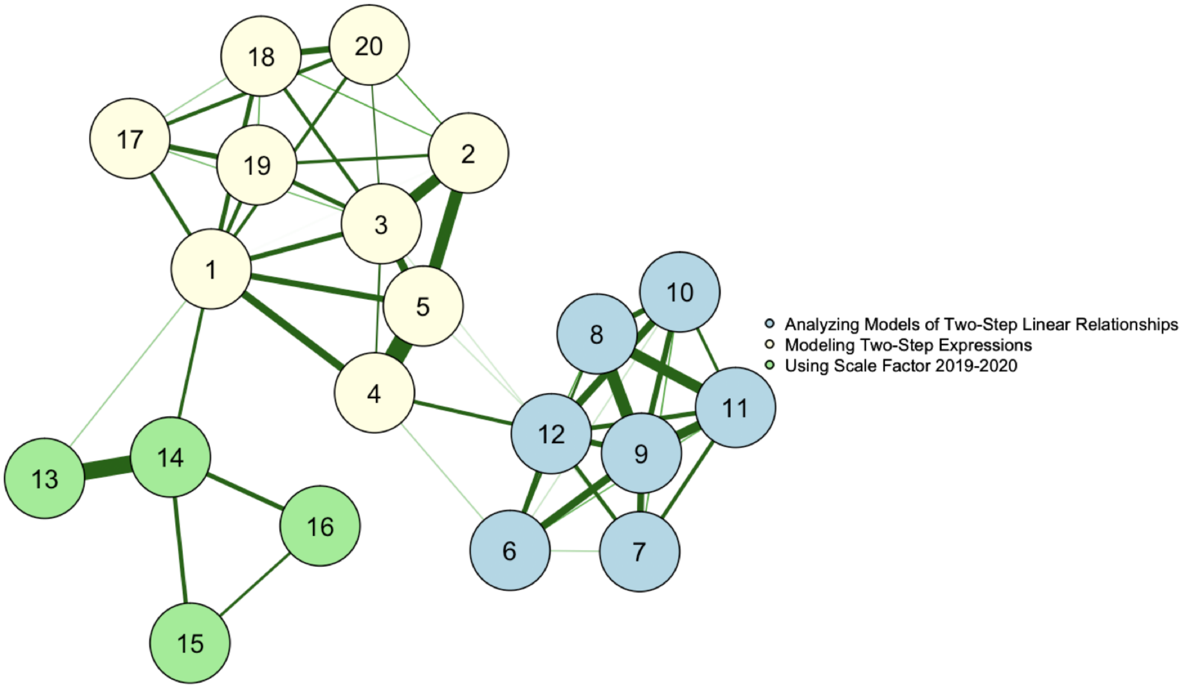
When the number of initial opportunities is 1:

```
ggm(dat = ggm_opp,num = 1,min_cor = 0.05)
```


18
20
13
17
1
2
14
3
19
15
5
16
4
Analyzing Models of Two-Step Linear Relationships
Modeling Two-Step Expressions
Using Scale Factor 2019-2020
8
7
9
10
11
6
12

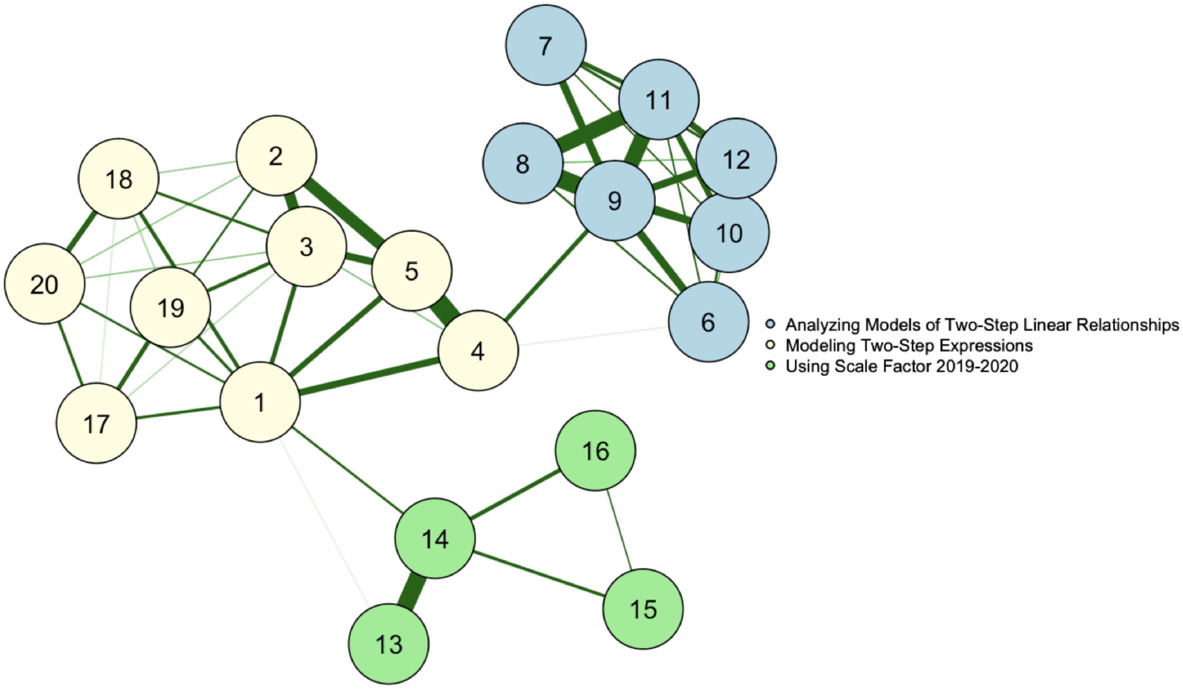
When the number of initial opportunities is 2:

```
ggm(dat = ggm_opp,num = 2,min_cor = 0.05)
```


18
20
17
19
2
3
1
5
4
10
8
11
12
9
6
7
13
14
16
15
○ Analyzing Models of Two-Step Linear Relationships
○ Modeling Two-Step Expressions
○ Using Scale Factor 2019-2020

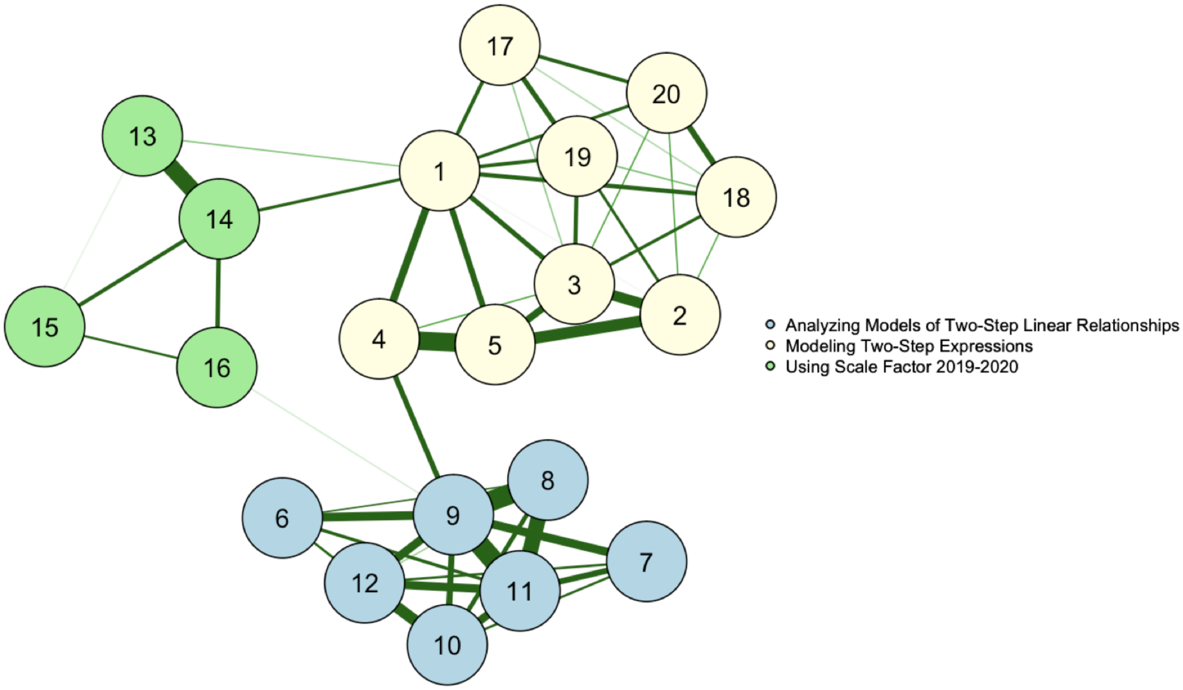
When the number of initial opportunities is 3

```
ggm(dat = ggm_opp,num = 3,min_cor = 0.05)
```


7
11
8
12
9
10
6
2
18
3
5
20
19
4
1
17
16
14
15
13
Analyzing Models of Two-Step Linear Relationships
Modeling Two-Step Expressions
Using Scale Factor 2019-2020

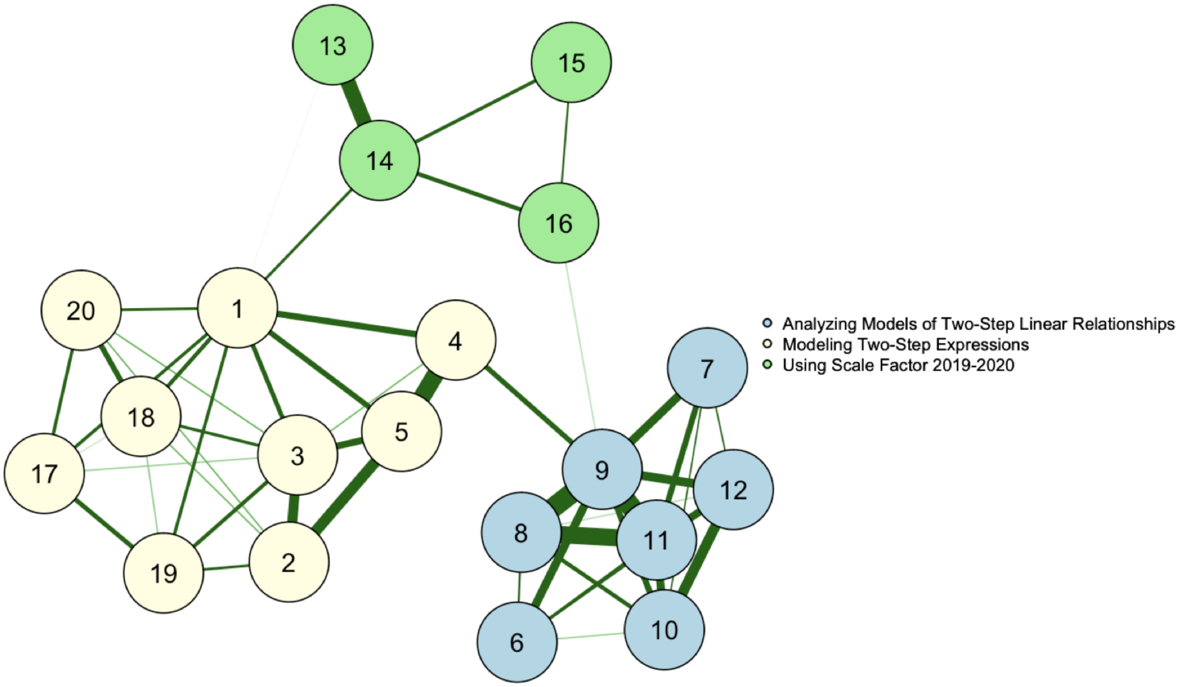
When the number of initial opportunities is 4

```
ggm(dat = ggm_opp,num = 4,min_cor = 0.05)
```


17
20
13
1
19
18
14
3
2
15
4
5
16
8
6
9
7
12
11
10
Analyzing Models of Two-Step Linear Relationships
Modeling Two-Step Expressions
Using Scale Factor 2019-2020

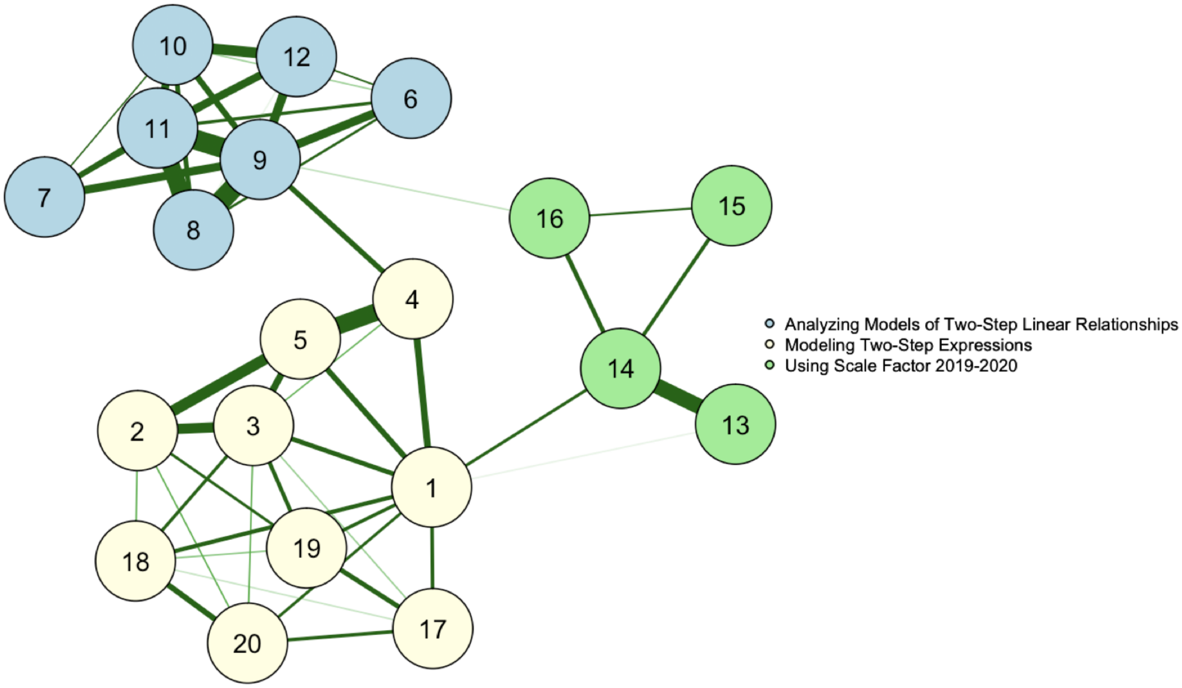
When the number of initial opportunities is 5

```
ggm(dat = ggm_opp,num = 5,min_cor = 0.05)
```


13
15
14
16
20
1
4
7
18
5
3
9
12
17
8
11
2
19
6
10
Analyzing Models of Two-Step Linear Relationships
Modeling Two-Step Expressions
Using Scale Factor 2019-2020

When the number of initial opportunities is 6

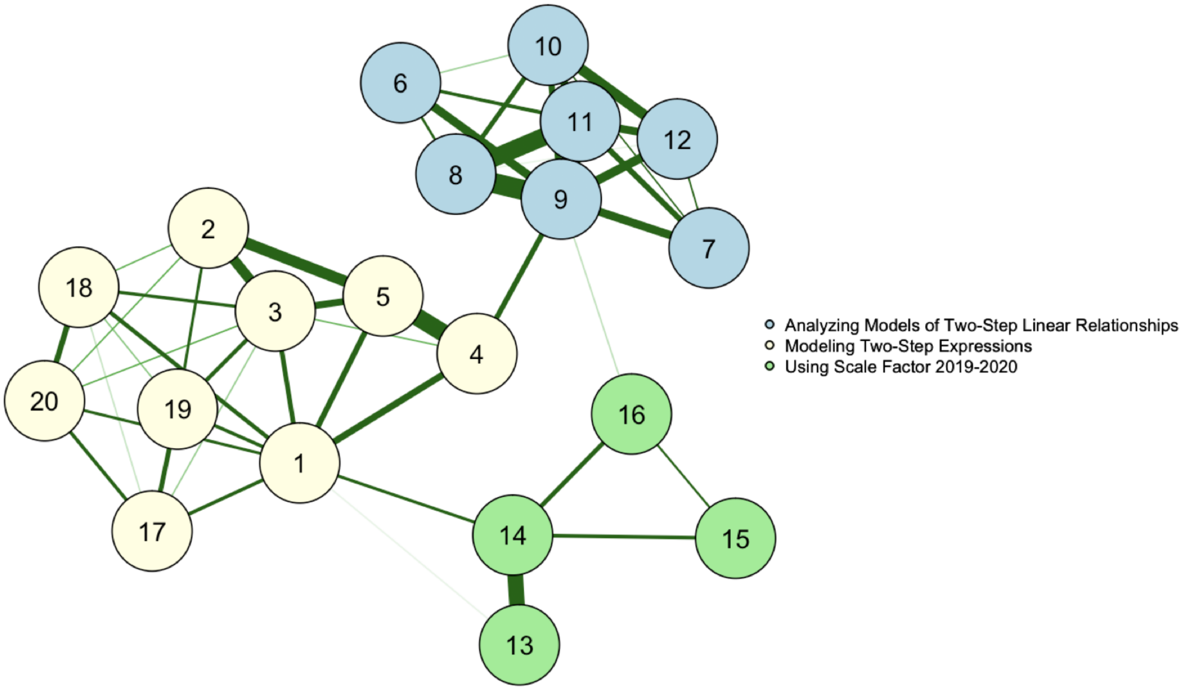
```
ggm(dat = ggm_opp,num = 6,min_cor = 0.05)
```



10
12
6
11
9
7
8
16
15
4
5
14
2
3
13
1
18
19
17
20
Analyzing Models of Two-Step Linear Relationships
Modeling Two-Step Expressions
Using Scale Factor 2019-2020

When the number of initial opportunities is 7

```
ggm(dat = ggm_opp,num = 7,min_cor = 0.05)
```


10
6
11
12
8
9
7
2
18
3
5
4
20
19
1
17
16
14
15
13
● Analyzing Models of Two-Step Linear Relationships
● Modeling Two-Step Expressions
● Using Scale Factor 2019-2020

As we can see, when the number of initial opportunities is 2, the partial correlations between workspace A and workspace B are the strongest. There are more lines between those two workspaces. Therefore, we chose 2 as our final cutoff point for initial opportunities.

**Appendix 3. Mixed Effects Logistic Regression**

## Initial Data Processing

```
library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3     v purrr   0.3.4

## v tibble  3.0.4     v dplyr   1.0.2

## v tidyr   1.1.2     v stringr 1.4.0

## v readr   1.4.0     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --

## x dplyr::filter() masks stats::filter()

## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(lme4)
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

Read in workspace B data.

```
b = read.delim("b.txt", header = T)
```

Only keep records where KC is not null

```
b$KC.Model.MATHia. = ifelse(b$KC.Model.MATHia. == "", NA, b$KC.Model.MATHia.)
b = b[!is.na(b$KC.Model.MATHia.),]
```

Get rid of unneccessary columns

```
b_clean = b[, c(1, 3, 5, 6, 8, 10, 11, 12, 13, 17)]
```

Read in workspace B step rollup data, keep records where KC is not null

```
b_student = read.delim("b_student.txt", header = T)
b_student$KC..MATHia. = ifelse(b_student$KC..MATHia. == "",
                               NA, b_student$KC..MATHia.)
b_student = b_student[!is.na(b_student$KC..MATHia.),]
```

Get rid of unneccessary columns and recode first attempt column

```
b_student_clean = b_student[, c(3, 4, 5, 7, 8, 15, 20, 21)]
b_student_clean$First.Attempt = ifelse(b_student_clean$First.Attempt ==
                               "correct", 1, 0)
```

## Modeling

There are 9 unique KCs in workspace B

```
unique(b_student_clean$KC..MATHia.)

## [1] "identifying units-1"

## [2] "enter given, reading numerals-1"

## [3] "define variable-1"

## [4] "find y, any form-1"

## [5] "write expression, positive intercept-1"

## [6] "enter given, reading words-1"

## [7] "write expression, negative slope-1"

## [8] "write expression, positive slope-1"

## [9] "write expression, negative intercept-1"
```

Merge two datasets and add indicators of whether student has mastered KC1 and KC2

```
kc.1 = "identifying units-1"

kc.2 = "write expression, negative slope-1"


count = 0

count.1 = 0


df_all = NA


for (id in unique(b_student_clean$Anon.Student.Id)){

  temp.1 = b_clean[which(b_clean$Anon.Student.Id == id &

                         b_clean$KC.Model.MATHia.%in% c(kc.1, kc.2) &

                         b_clean$Attempt.At.Step == 1), ]

  temp.1 = unique(temp.1)

  temp.1 = temp.1[order(temp.1$Time), ]
  temp.1 = unique(temp.1[, -2])

  temp.2 = b_student_clean[which(b_student_clean$Anon.Student.Id == id &
                         b_student_clean$KC..MATHia. %in% c(kc.1, kc.2)), ]

  if (nrow(temp.1) != nrow(temp.2)){
      count = count + 1
      next
  }

  temp.2$CF = temp.1$CF..Skill.New.p.Known.

  df = temp.2[, c(1, 6, 7, 8, 9)]
  df$CF.ind = ifelse(df$CF > 0.95, 1, 0)

  if (length(which(df$KC..MATHia. == kc.2 & df$CF.ind == 1)) == 0){
      count.1 = count.1 + 1
      next
  }
  if (length(which(df$KC..MATHia. == kc.1 & df$CF.ind == 1)) == 0){
      count.1 = count.1 + 1
      next
  }

  if (which(df$KC..MATHia. == kc.2 & df$CF.ind == 1)[1]-1 == 0){
      df$know_kc2 = rep(1, nrow(df)+1-which(df$KC..MATHia. == kc.2 & df$CF.ind == 1)[1])
  }
  else{
          df$know_kc2 = c(rep(0, which(df$KC..MATHia. == kc.2 & df$CF.ind == 1)[1]-1),
                  rep(1, nrow(df)+1-which(df$KC..MATHia. == kc.2 & df$CF.ind == 1)[1]))
  }

  if (which(df$KC..MATHia. == kc.1 & df$CF.ind == 1)[1]-1 == 0){
          df$know_kc1 = rep(1, nrow(df)+1-which(df$KC..MATHia. == kc.1 & df$CF.ind == 1)[1])
  }
  else{
          df$know_kc1 = c(rep(0, which(df$KC..MATHia. == kc.1 & df$CF.ind == 1)[1]-1),
                  rep(1, nrow(df)+1-which(df$KC..MATHia. == kc.1 & df$CF.ind == 1)[1]))
  }

  if (is.na(df_all)){
          df_all = df
  }
  else{
          df_all = rbind(df_all, df)
  }

  }
```

Run Glmer on KC1 and KC2

```
df.kc1 = df_all[which(df_all$KC..MATHia. == kc.1), ]
fit.7 = glmer(First.Attempt ~ 1 + Opportunity..MATHia. +
        Opportunity..MATHia. : know_kc2 + know_kc2 + (1|Anon.Student.Id),
        data = df.kc1, family = "binomial")



df.kc2 = df_all[which(df_all$KC..MATHia. == kc.2), ]
fit.8 = glmer(First.Attempt ~ 1 + Opportunity..MATHia. +
        Opportunity..MATHia. : know_kc1 + know_kc1 + (1|Anon.Student.Id),
        data = df.kc2, family = "binomial")

summary(fit.7)
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula:
## First.Attempt ~ 1 + Opportunity..MATHia. + Opportunity..MATHia.:know_kc2 +
##     know_kc2 + (1 | Anon.Student.Id)
##    Data: df.kc1
##
##      AIC      BIC   logLik deviance df.resid
##    469.2    489.3   -229.6    459.2      402
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.4113 -0.8943  0.4520  0.6388  1.3639
##
## Random effects:
```

```
##  Groups        Name               Variance Std.Dev.
##  Anon.Student.Id (Intercept) 0.3837   0.6194
## Number of obs: 407, groups:  Anon.Student.Id, 31
##
## Fixed effects:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -0.29375           0.26232  -1.120   0.2628
## Opportunity..MATHia.           0.188660.03856   4.893 9.92e-07 ***
## know_kc2                           0.609730.69012   0.884   0.3770
## Opportunity..MATHia.:know_kc2 -0.10467    0.05234  -2.000   0.0455 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##         (Intr) Op..MATH. knw_k2
## Oppr..MATH. -0.740
## know_kc2     -0.267  0.163
## O..MATH.:_2  0.487 -0.584   -0.799
```

```
summary(fit.8)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula:
## First.Attempt ~ 1 + Opportunity..MATHia. + Opportunity..MATHia.:know_kc1 +
##         know_kc1 + (1 | Anon.Student.Id)
##         Data: df.kc2
```

```
##
##        AIC      BIC   logLik deviance df.resid
##        101.7    113.5    -45.8     91.7      74
##
## Scaled residuals:
##        Min      1Q  Median      3Q       Max
## -1.9618 -0.9060  0.3901  0.6027  1.2727
##
## Random effects:
##  Groups        Name               Variance Std.Dev.
##  Anon.Student.Id (Intercept) 1.042     1.021
## Number of obs: 79, groups:  Anon.Student.Id, 31
##
## Fixed effects:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -0.1301             1.5395  -0.085   0.933
## Opportunity..MATHia.          0.4384  1.2311   0.356   0.722
## know_kc1                           1.8751  1.6920   1.108   0.268
## Opportunity..MATHia.:know_kc1  -0.4384     1.2116  -0.362   0.717
##
## Correlation of Fixed Effects:
##        (Intr) Op..MATH. knw_k1
## Oppr..MATH. -0.931
## know_kc1     -0.902  0.855
## O..MATH.:_1  0.929 -0.975   -0.898
```