1. Using the spreadsheets that we used in Exercise 2, import the xlsx files into Rstudio.

setwd("/Users/troygloyn/desktop/DHDA/Data Science and Analytics I")

Comment = read\_excel("Comment2.xlsx", col\_names = TRUE)

FallTable = read\_excel("FallTable2.xlsx", col\_names = TRUE)

InterviewTable = read\_excel("InterviewTable2.xlsx", col\_names = TRUE)

MedicalHistory = read\_excel("MedicalHistory2.xlsx", col\_names = TRUE)

> setwd("/Users/troygloyn/desktop/DHDA/Data Science and Analytics I")

>

> Comment = read\_excel("Comment2.xlsx", col\_names = TRUE)

> FallTable = read\_excel("FallTable2.xlsx", col\_names = TRUE)

> InterviewTable = read\_excel("InterviewTable2.xlsx", col\_names = TRUE)

> MedicalHistory = read\_excel("MedicalHistory2.xlsx", col\_names = TRUE)

1. Sort each of them by SubjectID

Comment\_sort = Comment[order(Comment$SubjectID), ]

FallTable\_sort = FallTable[order(FallTable$SubjectID),]

InterviewTable\_sort = InterviewTable[order(InterviewTable$SubjectID),]

MedicalHistory\_sort = MedicalHistory[order(MedicalHistory$SubjectID),]

> Comment\_sort = Comment[order(Comment$SubjectID), ]

> FallTable\_sort = FallTable[order(FallTable$SubjectID),]

> InterviewTable\_sort = InterviewTable[order(InterviewTable$SubjectID),]

> MedicalHistory\_sort = MedicalHistory[order(MedicalHistory$SubjectID),]

1. Merge all of them together in an **inner join** as a new dataframe

Merge1 = merge(x=Comment\_sort, y=FallTable\_sort, by="SubjectID")

Merge2 = merge(Merge1, InterviewTable\_sort, by="SubjectID")

AllTables = merge(Merge2, MedicalHistory\_sort, by="SubjectID")

> Merge1 = merge(x=Comment\_sort, y=FallTable\_sort, by="SubjectID")

> Merge2 = merge(Merge1, InterviewTable\_sort, by="SubjectID")

> AllTables = merge(Merge2, MedicalHistory\_sort, by="SubjectID")

1. Calculate time SINCE the interview date (SIDateiv) - using your "today's date" (whatever it may be) – express the answer in months (assign this to a variable called timesinceint)

AllTables$timesinceint = difftime(Sys.Date(), AllTables$SIDateiv, units = c("days"))

AllTables$timesinceint = round(as.numeric(AllTables$timesinceint)/29.53,0)

> AllTables$timesinceint = difftime(Sys.Date(), AllTables$SIDateiv, units = c("days"))

> AllTables$timesinceint = round(as.numeric(AllTables$timesinceint)/29.53,0)

1. Subset this dataframe by selecting only cases where timesinceint is >= mean of the value of timesinceint across all participants (e.g. the 50% most recent participants)

mean\_int = mean(AllTables$timesinceint)

AllTables2 = AllTables[AllTables$timesinceint >= mean\_int,]

> mean\_int = mean(AllTables$timesinceint)

> AllTables2 = AllTables[AllTables$timesinceint >= mean\_int,]

1. Create another subset of this dataframe from 5) selecting only:  **SubjectID**and these variables:

**SDOsteoDiag      SDRheumaDiag SDOADiag            SDLupusDiag      SDHyperthyDiag**

AllTables3 = AllTables2[,c("SubjectID","SDOsteoDiag","SDRheumaDiag","SDOADiag","SDLupusDiag","SDHyperthyDiag")]

> AllTables3 = AllTables2[,c("SubjectID","SDOsteoDiag","SDRheumaDiag","SDOADiag","SDLupusDiag","SDHyperthyDiag")]

1. Use the mutate function to recode the yes and no's to 1's and 0's, respectively. Also recode missing data and Don't know to NA (the actual missing data denotation in R, not the string "NA").

AllTables3 = mutate(AllTables3, SDOsteoDiag=ifelse(SDOsteoDiag=="Yes",1,ifelse(SDOsteoDiag=="No",0,SDOsteoDiag)))

AllTables3 = mutate(AllTables3, SDRheumaDiag=ifelse(SDRheumaDiag=="Yes",1,ifelse(SDRheumaDiag=="No",0,SDRheumaDiag)))

AllTables3 = mutate(AllTables3, SDOADiag=ifelse(SDOADiag=="Yes",1,ifelse(SDOADiag=="No",0,SDOADiag)))

AllTables3 = mutate(AllTables3, SDLupusDiag=ifelse(SDLupusDiag=="Yes",1,ifelse(SDLupusDiag=="No",0,SDLupusDiag)))

AllTables3 = mutate(AllTables3, SDHyperthyDiag=ifelse(SDHyperthyDiag=="Yes",1,ifelse(SDHyperthyDiag=="No",0,SDHyperthyDiag)))

AllTables3 = mutate(AllTables3, SDOsteoDiag=ifelse(SDOsteoDiag=="Don't Know" | SDOsteoDiag=="missing data",NA,SDOsteoDiag))

AllTables3 = mutate(AllTables3, SDRheumaDiag=ifelse(SDRheumaDiag=="Don't Know" | SDRheumaDiag=="missing data",NA,SDRheumaDiag))

AllTables3 = mutate(AllTables3, SDOADiag=ifelse(SDOADiag=="Don't Know" | SDOADiag=="missing data",NA,SDOADiag))

AllTables3 = mutate(AllTables3, SDLupusDiag=ifelse(SDLupusDiag=="Don't Know" | SDLupusDiag=="missing data",NA,SDLupusDiag))

AllTables3 = mutate(AllTables3, SDHyperthyDiag=ifelse(SDHyperthyDiag=="Don't Know" | SDHyperthyDiag=="missing data",NA,SDHyperthyDiag))

> AllTables3 = AllTables2[,c("SubjectID","SDOsteoDiag","SDRheumaDiag","SDOADiag","SDLupusDiag","SDHyperthyDiag")]

> AllTables3 = mutate(AllTables3, SDOsteoDiag=ifelse(SDOsteoDiag=="Yes",1,ifelse(SDOsteoDiag=="No",0,SDOsteoDiag)))

> AllTables3 = mutate(AllTables3, SDRheumaDiag=ifelse(SDRheumaDiag=="Yes",1,ifelse(SDRheumaDiag=="No",0,SDRheumaDiag)))

> AllTables3 = mutate(AllTables3, SDOADiag=ifelse(SDOADiag=="Yes",1,ifelse(SDOADiag=="No",0,SDOADiag)))

> AllTables3 = mutate(AllTables3, SDLupusDiag=ifelse(SDLupusDiag=="Yes",1,ifelse(SDLupusDiag=="No",0,SDLupusDiag)))

> AllTables3 = mutate(AllTables3, SDHyperthyDiag=ifelse(SDHyperthyDiag=="Yes",1,ifelse(SDHyperthyDiag=="No",0,SDHyperthyDiag)))

> AllTables3 = mutate(AllTables3, SDOsteoDiag=ifelse(SDOsteoDiag=="Don't Know" | SDOsteoDiag=="missing data",NA,SDOsteoDiag))

> AllTables3 = mutate(AllTables3, SDRheumaDiag=ifelse(SDRheumaDiag=="Don't Know" | SDRheumaDiag=="missing data",NA,SDRheumaDiag))

> AllTables3 = mutate(AllTables3, SDOADiag=ifelse(SDOADiag=="Don't Know" | SDOADiag=="missing data",NA,SDOADiag))

> AllTables3 = mutate(AllTables3, SDLupusDiag=ifelse(SDLupusDiag=="Don't Know" | SDLupusDiag=="missing data",NA,SDLupusDiag))

> AllTables3 = mutate(AllTables3, SDHyperthyDiag=ifelse(SDHyperthyDiag=="Don't Know" | SDHyperthyDiag=="missing data",NA,SDHyperthyDiag))

1. Create a separate dataframe just like in 7) and try to recode again using the ifelse statement instead

AllTables4 = AllTables2[,c("SubjectID","SDOsteoDiag","SDRheumaDiag","SDOADiag","SDLupusDiag","SDHyperthyDiag")]

varlist = colnames(AllTables4)[2:6]

for(i in varlist){

AllTables4[[i]] = ifelse(AllTables4[[i]] == "Yes","1",ifelse(AllTables4[[i]] == "No", "0",

ifelse(AllTables4[[i]] == "Don't Know" | AllTables4[[i]] == "missing data", NA,AllTables4[[i]])))

}

AllTables4

> for(i in varlist){

+ AllTables4[[i]] = ifelse(AllTables4[[i]] == "Yes","1",ifelse(AllTables4[[i]] == "No", "0",

+ ifelse(AllTables4[[i]] == "Don't Know" | AllTables4[[i]] == "missing data", NA,AllTables4[[i]])))

+ }

> AllTables4

SubjectID SDOsteoDiag SDRheumaDiag SDOADiag SDLupusDiag SDHyperthyDiag

1 AM10353 0 0 1 0 0

2 AM10355 1 0 0 0 0

3 AM10357 1 0 0 0 0

4 AM10359 1 1 1 0 0

5 AM10361 0 0 0 0 0

6 AM10365 <NA> 1 0 <NA> 0

7 AM10375 0 0 1 0 0

8 AM10384 1 0 <NA> 0 0

9 AM10385 0 0 0 0 0

10 AM10386 0 1 0 0 0

11 AM10389 0 0 0 0 0

12 AM10391 0 0 1 0 0

13 AM10392 0 0 0 0 0

14 AM10393 1 0 0 0 0

15 AM10396 1 0 0 0 0

16 AM10399 0 0 0 0 0

17 AM10400 0 0 0 0 0

18 AM10401 0 0 1 0 0

19 AM10402 0 0 1 0 0

20 AM10404 0 0 0 0 0

21 AM10405 1 0 1 0 0

23 AM10408 0 0 0 0 0

24 AM10411 0 0 1 0 0

25 AM10413 0 0 1 0 0

26 AM10415 1 0 0 0 0

27 AM10418 0 0 0 0 0

28 AM10419 1 0 0 0 0

29 AM10420 0 0 0 0 0

30 AM10422 1 0 1 0 0

31 AM10425 1 1 1 0 0

32 AM10427 0 0 0 0 0

33 AM10429 0 0 0 0 0

34 AM10435 0 0 0 0 0

35 AM10436 1 0 1 0 0

36 AM10444 0 0 0 0 0

37 AM10451 0 0 1 0 0

38 AM10452 0 0 1 0 0

39 AM10455 0 0 1 0 0

40 AM10459 1 0 <NA> 0 0

41 AM10462 0 0 1 0 0

42 AM10465 0 0 0 0 0

43 AM10468 0 0 1 0 0

44 AM10470 0 1 0 0 0

45 AM10473 0 0 0 0 0

46 AM10476 0 0 0 0 0

47 AM10479 0 0 0 0 0

48 AM10480 0 0 1 0 0

49 AM10481 0 0 1 0 0

50 AM10483 0 0 0 0 0

51 AM10487 1 0 0 0 0

52 AM10488 0 0 1 0 0

53 AM10489 0 0 0 0 0

54 AM10490 1 0 1 0 1

55 AM10491 1 1 0 0 0

56 AM10492 0 0 1 0 0

57 AM10499 0 0 0 0 0

58 AM10501 1 0 0 0 0

59 AM10502 0 0 0 0 0

60 AM10507 1 0 0 0 0

61 AM10508 0 0 0 0 0

62 AM10509 1 0 0 0 0

63 AM10511 1 0 0 0 0

64 AM10515 0 0 1 0 0

65 AM10518 0 0 1 0 0

66 AM10519 0 0 1 0 0

67 AM10525 0 0 0 0 0

68 AM10528 0 0 0 0 0

69 AM10530 0 0 0 0 0

70 AM10531 0 0 1 0 0

71 AM10536 0 0 0 0 0

72 AM10544 1 0 1 0 0

73 AM10547 <NA> 0 0 0 0

74 AM10548 0 0 0 0 0

75 AM10555 1 0 0 0 0

76 AM10556 1 1 1 0 0

77 AM10557 1 1 0 0 0

78 AM10559 0 0 0 0 0

79 AM10562 0 0 1 0 0

80 AM10564 1 0 1 0 0

81 AM10565 1 1 1 0 0

82 AM10566 0 0 1 0 0

83 AM10571 1 0 0 0 0

84 AM10576 1 0 1 0 0

85 AM10578 0 0 1 0 0

86 AM10581 0 0 0 0 0

87 AM10582 0 0 1 0 1

88 AM10583 1 0 1 0 0

89 AM10584 1 0 1 0 0

90 AM10587 0 0 0 0 0

91 AM10589 1 0 1 0 0

92 AM10590 0 0 1 0 0

93 AM10594 0 0 0 0 1

94 AM10597 1 0 1 0 0

95 AM10601 0 0 0 0 0

96 AM10604 0 0 0 0 0

97 AM10606 0 0 0 0 0

98 AM10607 0 0 0 0 0

99 AM10609 0 0 1 0 0

100 AM10611 0 0 0 0 0

101 AM10612 1 1 0 0 0

102 AM10618 0 0 1 0 0

103 AM10622 1 1 1 0 0

104 AM10623 0 0 0 0 0

105 AM10624 0 0 0 0 0

106 AM10626 1 1 0 0 0

107 AM10632 1 0 0 0 0

108 AM10633 0 0 0 0 0

109 AM10634 0 0 1 0 0

110 AM10639 0 0 0 0 0

111 AM10641 0 0 0 0 0

112 AM10642 0 0 1 0 0

113 AM10643 0 0 0 0 0

114 AM10651 1 0 1 0 0

115 AM10653 0 0 0 0 0

117 AM10657 1 0 0 0 0

118 AM10658 <NA> <NA> <NA> <NA> <NA>

119 AM10661 1 0 0 0 0

120 AM10662 0 0 1 0 0

122 AM10668 0 0 1 0 0

123 AM10669 0 0 0 0 0

124 AM10671 0 0 0 0 0

125 AM10672 0 0 0 0 0

126 AM10673 0 <NA> 0 0 0

158 AM20353 0 0 0 0 0

159 AM20356 0 0 0 0 0

160 AM20357 0 0 0 0 0

161 AM20363 0 0 0 0 0

162 AM20365 0 0 0 0 0

163 AM20367 <NA> <NA> 1 <NA> <NA>

164 AM20372 0 0 0 0 0

165 AM20373 0 0 0 0 0

166 AM20374 0 0 1 0 0

167 AM20382 0 0 0 0 0

168 AM20385 1 0 1 0 0

169 AM20391 0 0 1 0 0

170 AM20392 0 0 0 0 0

171 AM20395 1 0 1 0 0

173 AM20401 0 0 0 0 0

175 AM20409 0 0 1 0 0

179 AM20418 0 0 1 0 0

181 AM20425 0 0 1 0 0

183 AM20427 1 0 1 0 0

184 AM20429 0 0 1 0 0

186 AM20432 1 0 0 0 0

190 AM20440 0 0 0 0 0