

Mervar - Lab 4 (Pattern Recognition)

Alexander Mervar

2022-10-05

Provided Code

```
library(ez) #for doing the ANOVAs
library(dplyr) # great library for massaging data

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

library(ggplot2)

# I put the data file in a specific folder. If you don't have the same folder structures as me, you'll
data<-read.table("featureSearchResults.csv", stringsAsFactors=TRUE,sep=',', header = TRUE)
data<-filter(data,trial_type=="visual-search-circle") # only include actual feature search trials, not
data<-droplevels(data) #eliminate NULLS left over from instructions. Only include factor levels that a
sizes<-c(1,5,15,30) # create a list of the set sizes in their desired order
data$set_size<-factor(data$set_size, levels = sizes) #reorder the four levels of set_size so that they
table(data$subject_id) # make sure that everybody did the same number of trials

##
##      2000081652  abassard  abmapete  adecourc  afwaddle  agpanago
##      4800      400      400      400      400      400      400
##  airwheel  amervar   anafe  andbstew  anfreel  anikouli3  anykim
##      400      400      400      400      400      400      400
##   apiero  apmolina  arwingar   asm9  astilian  atimothy  axelrodb
##      400      400      400      400      400      400      400
##   beckad  beejones  behughey  billgree  bkwatts  brjoande  brownlk
##      400      400      400      400      400      400      400
##   bunnerj  cadakinc (  cashew  caudrets  cfridman  chlmcnai  cisch
##      400      400      400      400      400      400      400
##      cphi  dahaggar  Dian Zhi  ebooras  efenogli  Eliwebs  emnpayne
##      400      400      400      400      400      400      400
##  episrael  escain   ggball  gghurley   gjude   gnbond  greeneaj
##      400      400      400      400      400      400      400
##  grifhall  gripatel  gwaldow  hahmann  halamerr  hcorbin  hkrasnow
##      400      400      400      400      400      400      400
##   howargr  igeorgia  isdbord  isherfic  islynch  jackmosk  jaenalex
```

```
##      400      400      400      400      400      400      400
## jakemage jcworrel jermaldo jm122 Joe johnpurc josdoug1
##      400      400      400      400      400      400      400
## jotshiel jowamajo jsensanb jstamets jtadley kb kdobias
##      400      400      400      400      400      400      400
## Keiland khapatel kjkunkle knakajim lcgress lcwade lekowals
##      400      400      400      400      400      400      400
## lfaimon lgholmes liziyi Madison By madvogel marlshep2 Maxwell Va
##      400      400      400      400      400      400      800
## MAYGATES micsiler mirshelt mklutzke mlemos momamcgu moominpapa
##      400      400      400      400      400      400      10
## mrmornin2R msmwakal mvschnab mzelenin mzubi natcho nawhippo
##      400      400      400      400      400      400      400
## ngperkin nicgraha nlepstei novricha nwmaeda pauchamb philljoe
##      400      400      400      400      400      400      400
## pwsearle rajulian ramosjaa revankap rkellems rodashby ronnapie
##      400      400      400      400      400      400      400
## rpw rymfrase sakgoel samponde sarodri sdalmia sdings
##      400      400      400      400      400      400      400
## selzheng sercolli sidrath snwoodar spunday stffrenc sullisea
##      400      400      400      400      400      400      400
## sydsearc tdell thgibson tmdeckar tnperetz tyforajt tyoberbr
##      400      400      400      400      400      400      400
## vtlanglo weiyiy wmanneri yuhtang zegraber zhangjac zpetroff
##      400      400      400      400      400      400      400
## zrajwany zsdaniel
##      400      400
```

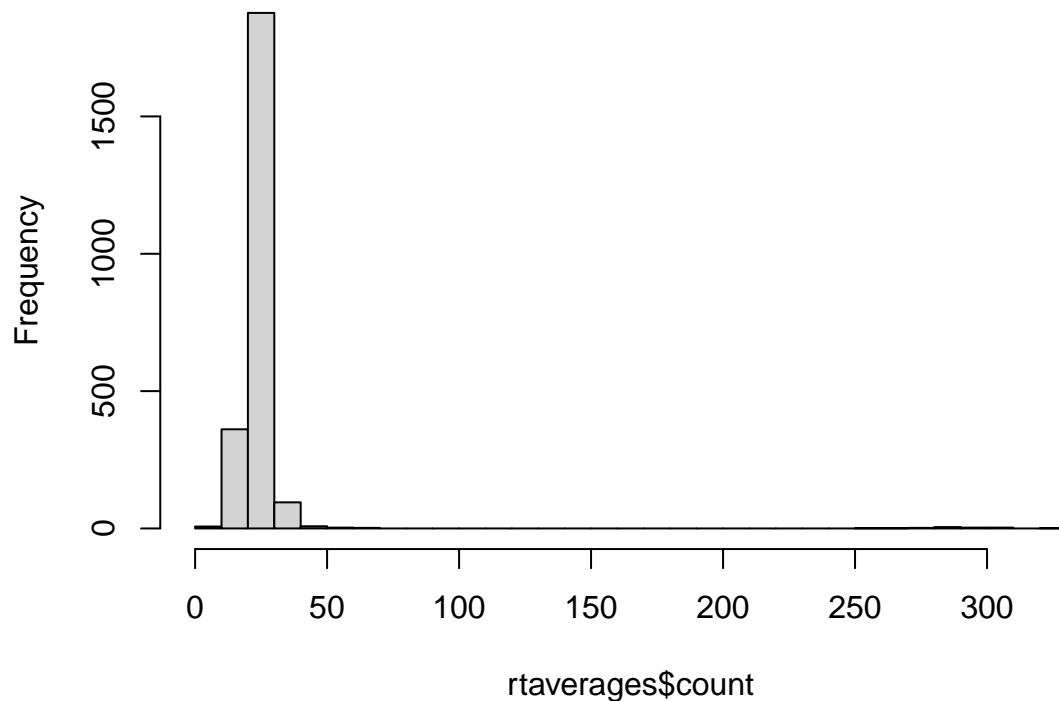
```
par(mar = c(4,4,4,4),mfrow=c(1, 1) ) # reset plots just in case it's necessary from whatever was being
```

```
#For looking at whether and how response time is influenced by the experimental variables
searchTimes<-filter(data, correct==1 ) # only include correct trials for response time analyses.
byseveral<-group_by(searchTimes,subject_id,target_presence,set_size,task) #break data down by subject,
rtaverages<-summarize(byseveral,count=n(),rt=mean(rt, trim=0.1),year=max(year)) #show averages for each
```

```
## `summarise()` has grouped output by 'subject_id', 'target_presence',
## 'set_size'. You can override using the `.groups` argument.
```

```
# View(rtaverages) # if you want to see the whole frame with everybody's averages
hist(rtaverages$count,breaks=40) #make sure that there's enough correct data from each of the cells
```

Histogram of rtaverages\$count



```
rtaverages[rtaverages$count<5,] # report subject_id and condition in which low counts are found. If ev
```

```
## # A tibble: 3 x 7
## # Groups:   subject_id, target_presence, set_size [3]
##   subject_id target_presence set_size task      count   rt   year
##   <fct>      <fct>          <fct> <fct>    <int> <dbl> <int>
## 1 astilian   present             30    disjunctive    2 1094. 2016
## 2 moominpapa present             5     disjunctive    4 5142 2018
## 3 novricha   present            15     disjunctive    1 1762 2018
```

```
unique(rtaverages$subject_id) # tells us how many subjects we have
```

```
##   [1] 2000081652 abassard abmapete adecourc afwaddle
##   [7] agpanago airwheel amervar anafe andbstew anfreel
##  [13] anikouli3 anykim apiero apmolina arwingar asm9
##  [19] astilian atimothy axelrodb becad beejones behughey
##  [25] billgree bkwatts brjoande brownlk bunnerj cadakinc (
##  [31] cashew caudrets cfridman chlmcnai cisch cphi
##  [37] dahaggar Dian Zhi ebooras efenogli Eliwebs emnpayne
##  [43] episrael escain ggball gghurley gjude gnbond
##  [49] greeneaj grifhall gripatel gwaldow hahmann halamerr
##  [55] hcorbin hkrasnow howargr igeorgia isdbord isherfic
##  [61] islynch jackmosk jaenalex jakemage jcworrel jermaldo
##  [67] jm122 Joe johnpurc josdoug1 jotshiel jowamajo
##  [73] jsensanb jstamets jtadley kb kdobias Keiland
##  [79] khapatel kjkunkle knakajim lcgress lcwade lekowals
##  [85] lfaimon lgholmes liziya Madison By madvogel marlshep2
##  [91] Maxwell Va MAYGATES micsiler mirshelt mklutzke mlemos
##  [97] momamcgu moominpapa mrmornin2R msmwakal mvschnab mzelenin
```

```
## [103] mzubini natcho nawhippo ngperkin nicgraha nlepstei
## [109] novricha nwmaeda pauchamb philljoe pwsearle rajulian
## [115] ramosjaa revankap rkellems rodashby ronnapie rpw
## [121] rymfrase sakgoel samponde sarodri sdalmia sdings
## [127] selzheng sercolli sidrath snwoodar spunday stffrenc
## [133] sullisea sydsearc tdell thgibson tmdeckar tnperez
## [139] tyforajt tyoberbr vtlanglo weiyiy wmanneri yuhtang
## [145] zegraber zhangjac zpetroff zrajwany zsdaniel
## 149 Levels: 2000081652 abassard abmapete adecourc afwaddle ... zsdaniel
```

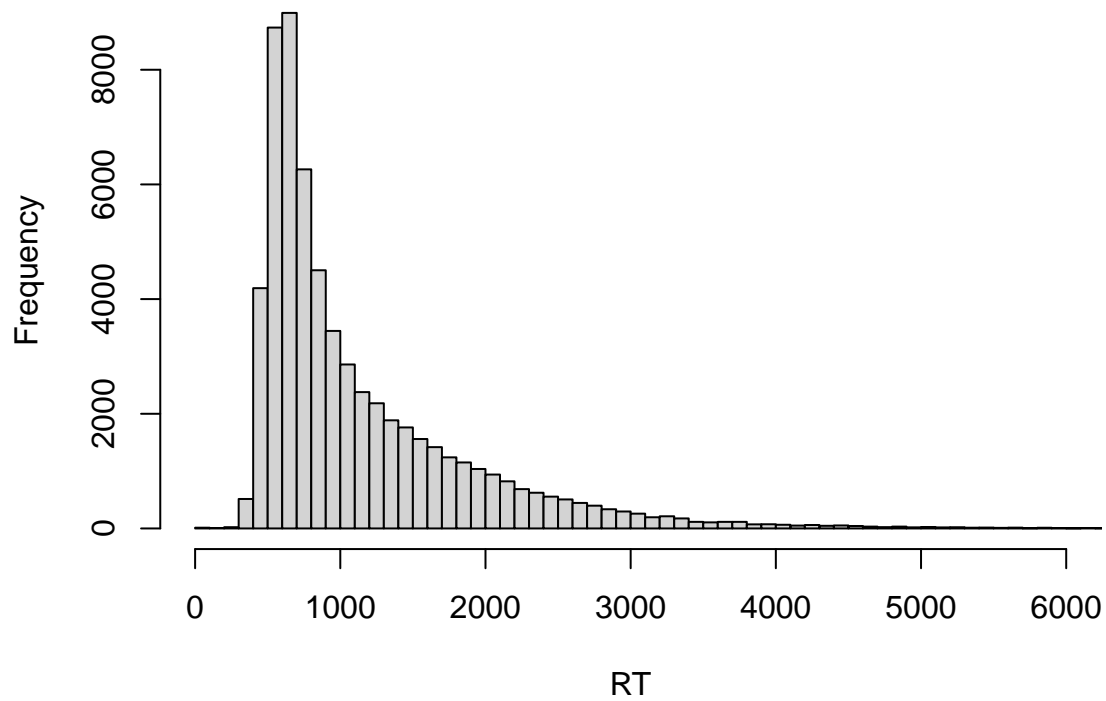
```
# cleaning of data - throw out "bad" subjects
```

```
unusualsubjects <- rtaverages$subject_id[rtaverages$count < 5 | rtaverages$count > 60] # make a list of
rtaverages <- filter(rtaverages,!(subject_id %in% unusualsubjects)) # only include data from good subjects
rtaverages<-droplevels(rtaverages) #drops levels (subjects) that are no longer represented in data after
unique(rtaverages$subject_id) # tells us how many remaining subjects we have
```

```
## [1] 2000081652 abassard abmapete adecourc afwaddle agpanago
## [7] airwheel amervar anafe andbstew anfreel anikouli3
## [13] anykim apiero apmolina arwingar asm9 atimothy
## [19] axelrodb becad beejones behughey billgree bkwatts
## [25] brjoande brownlk bunnerj cadakinc ( cashew caudrets
## [31] cfridman chlmcnai cisch cphi dahaggar Dian Zhi
## [37] ebooras efenogli Eliwebs emnpayne episrael escain
## [43] ggball gghurley gjude gnbond greeneaj grifhall
## [49] gripatel gwaldow hahmann halamerr hcorbin hkrasnow
## [55] howargr igeorgia isdbord isherfic islynch jackmosk
## [61] jaenalex jakemage jcworrel jermaldo jm122 Joe
## [67] johnpurc josdoug1 jotshiel jowamajo jsensanb jstamets
## [73] jtadley kb kdobias Keiland khapatel kjkunkle
## [79] knakajim lcgress lcwade lekowals lfaimon lgholmes
## [85] liziyi Madison By madvogel marlshep2 MAYGATES micsiler
## [91] mirshelt mklutzke mlemos momamcgu mrmornin2R msmwakal
## [97] mvschnab mzelenin mzubini natcho nawhippo ngperkin
## [103] nicgraha nlepstei nwmaeda pauchamb philljoe pwsearle
## [109] rajulian ramosjaa revankap rkellems rodashby ronnapie
## [115] rpw rymfrase sakgoel samponde sarodri sdalmia
## [121] sdings selzheng sercolli sidrath snwoodar spunday
## [127] stffrenc sullisea sydsearc tdell thgibson tmdeckar
## [133] tnperez tyforajt tyoberbr vtlanglo weiyiy wmanneri
## [139] yuhtang zegraber zhangjac zpetroff zrajwany zsdaniel
## 144 Levels: 2000081652 abassard abmapete adecourc afwaddle agpanago ... zsdaniel
```

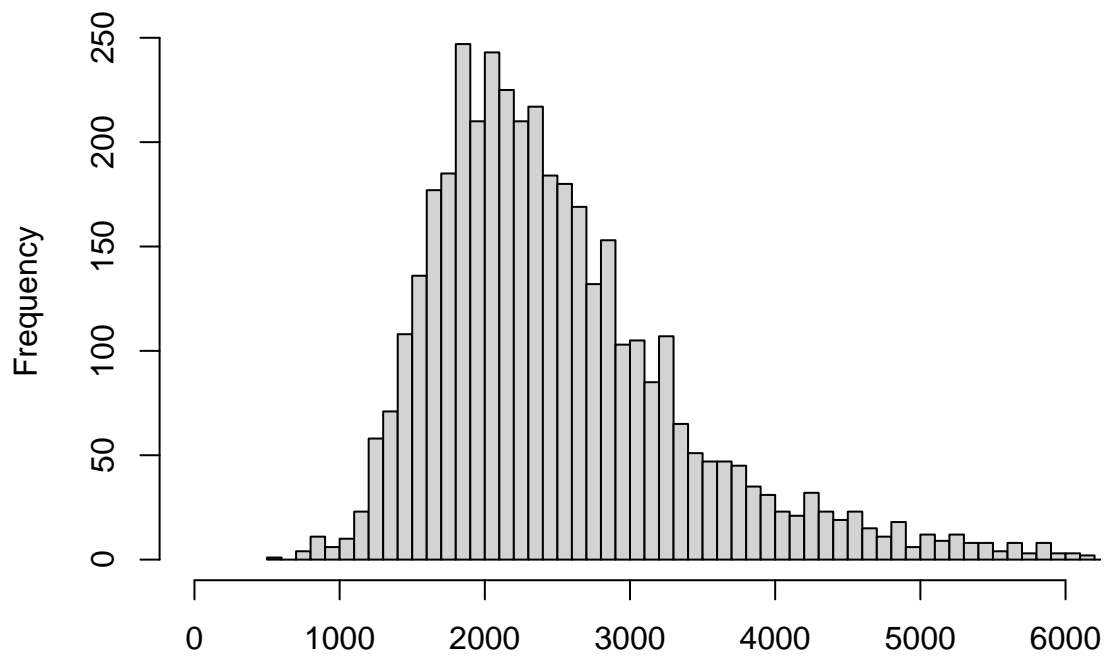
```
goodData<-filter(data,!(subject_id %in% unusualsubjects)) # make a data frame that only includes data from
goodData<-droplevels(goodData) #eliminate levels of dropped subjects
goodSearchTimes<-filter(goodData,correct==1) # for RT analysis, it is typical to only look at correct trials
hist(searchTimes$rt,breaks=200,xlim=c(0,6000),xlab="RT",ylab="Frequency") # get an overall feel for the
```

Histogram of searchTimes\$rt



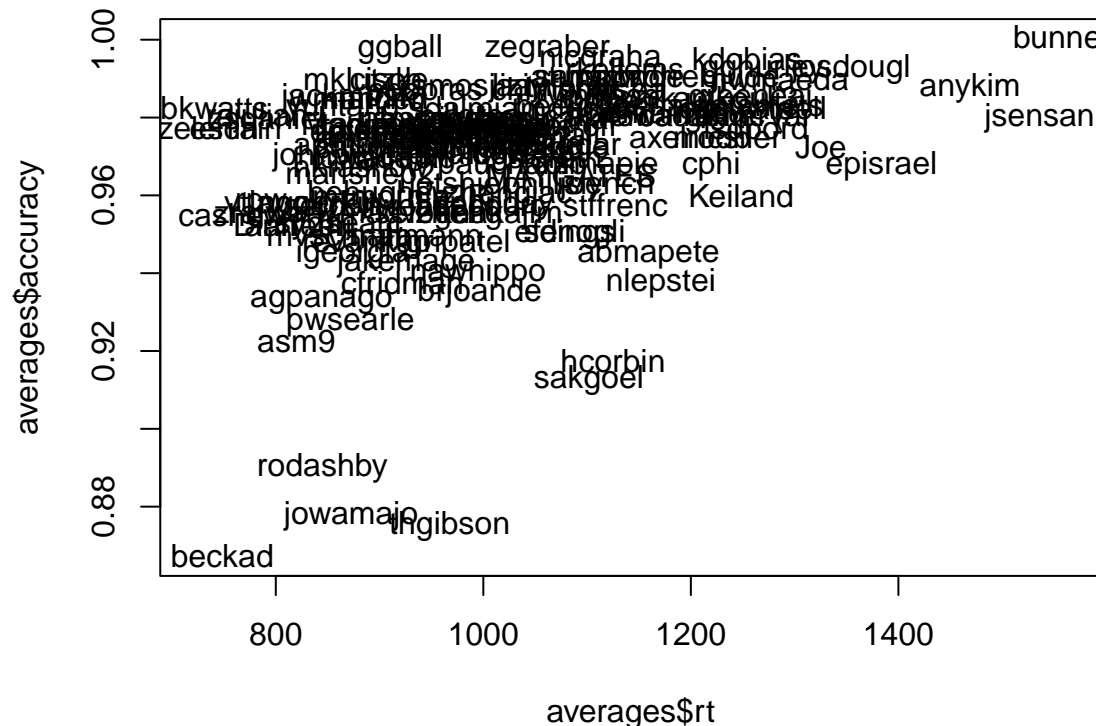
```
hist(filter(searchTimes, task=="conjunctive" & set_size=="30" & target_presence=="absent")$rt, breaks=200
```

searchTimes, task == "conjunctive" & set_size == "30" & target_preser



```
rt(searchTimes, task == "conjunctive" & set_size == "30" & target_presence == "absent")
```

```
# Below is code to look overall at speed and accuracy of each subject
bysubject<-group_by(goodData,subject_id) # break down data by subject - notice that we're using ALL data
averages<-summarize(bysubject,rt=mean(rt,trim=0.1),accuracy=sum(correct=="1")/n(),year=max(year)) #for
plot(averages$rt,averages$accuracy,type="n") # is there a speed-accuracy tradeoff? As people get faster
text(averages$rt,averages$accuracy,averages$subject_id)
```



```
cor.test(averages$rt,averages$accuracy) # is there a speed-accuracy tradeoff? Tends to be positive - a
```

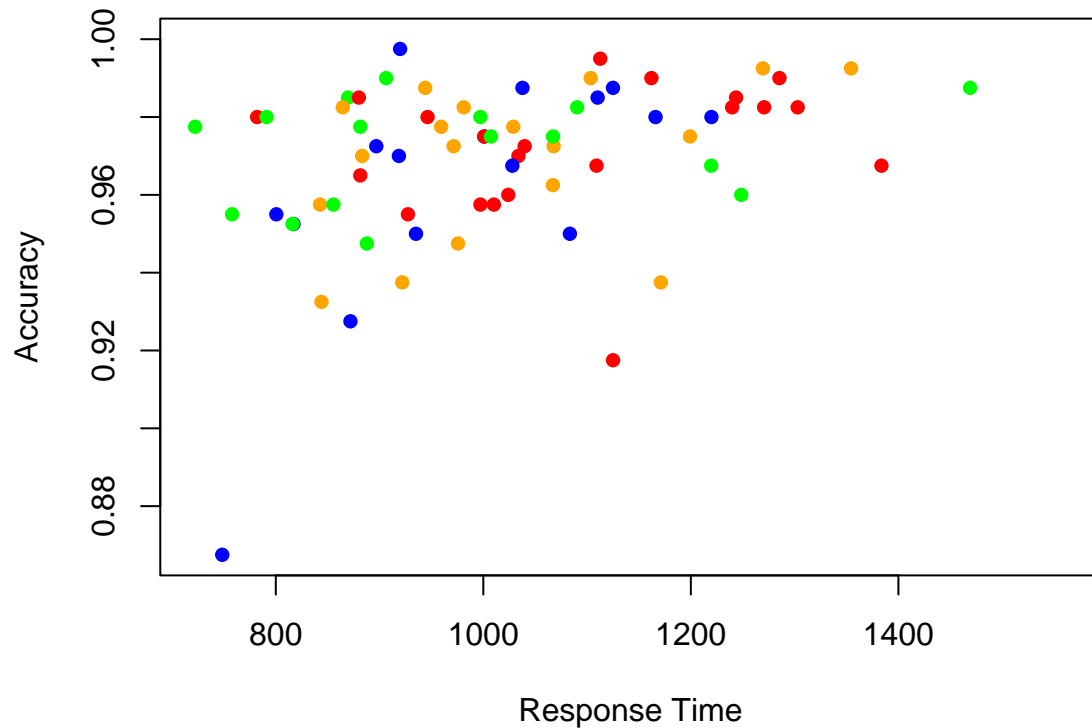
```
##
## Pearson's product-moment correlation
##
## data: averages$rt and averages$accuracy
## t = 4.3971, df = 142, p-value = 2.136e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.1935670 0.4824387
## sample estimates:
## cor
## 0.3461818
```

```
averages$color<-recode(averages$year,"2016"="red","2017"="orange","2018"="green","2019"="blue") # Dplyr
```

```
## Warning: Unreplaced values treated as NA as `x` is not compatible.
```

```
## Please specify replacements exhaustively or supply `.default`.
```

```
plot(averages$rt,averages$accuracy,pch=16,col=averages$color,xlab="Response Time",ylab="Accuracy") # is
```



```
table1 <- tapply(X=rtaverages$rt, INDEX=list(rtaverages$set_size, rtaverages$target_presence, rtaverages$task),
# table1 # show means so that one can begin to interpret the data
group_by(goodSearchTimes, set_size, target_presence, task) %>% summarize(mean(rt, trim=0.1)) # an alternative
```

```
## `summarise()` has grouped output by 'set_size', 'target_presence'. You can
## override using the `.groups` argument.
```

```
## # A tibble: 16 x 4
## # Groups:   set_size, target_presence [8]
##   set_size target_presence task      `mean(rt, trim = 0.1)`
##   <fct>    <fct>          <fct>          <dbl>
## 1 1      absent        conjunctive        601.
## 2 1      absent        disjunctive        633.
## 3 1      present       conjunctive        544.
## 4 1      present       disjunctive        613.
## 5 5      absent        conjunctive        827.
## 6 5      absent        disjunctive        897.
## 7 5      present       conjunctive        703.
## 8 5      present       disjunctive        688.
## 9 15     absent        conjunctive       1529.
## 10 15    absent        disjunctive       1637.
## 11 15    present       conjunctive       1054.
## 12 15    present       disjunctive        761.
## 13 30    absent        conjunctive       2405.
## 14 30    absent        disjunctive       2040.
## 15 30    present       conjunctive       1519.
## 16 30    present       disjunctive        847.
```

```
head(goodSearchTimes)
```

```
##   trial_index  rt subject_id      trial_type key_press set_size correct
## 1           2 1712   yuhtang visual-search-circle      70         1         1
```

```
## 2      3 741    yuhtang visual-search-circle      74      1      1
## 3      4 566    yuhtang visual-search-circle      74      1      1
## 4      5 998    yuhtang visual-search-circle      70      1      1
## 5      6 579    yuhtang visual-search-circle      70      1      1
## 6      7 803    yuhtang visual-search-circle      74      1      1
## target_presence task id year
## 1      absent conjunctive 3 2017
## 2      present conjunctive 4 2017
## 3      present conjunctive 5 2017
## 4      absent conjunctive 6 2017
## 5      absent conjunctive 7 2017
## 6      present conjunctive 8 2017
```

Fill in missing chunks

#Code to conduct actual ANOVA of the Response Times results

```
model<-ezANOVA(data=goodSearchTimes,dv=rt,within=c(set_size,target_presence,task),wid=subject_id) # You
```

```
## Warning: Collapsing data to cell means. *IF* the requested effects are a subset
## of the full design, you must use the "within_full" argument, else results may be
## inaccurate.
```

could include year as a variable with "between=year". Main effect maybe but would hope there wouldn't

```
model # show results of the ANOVA model
```

```
## $ANOVA
##              Effect DFn DFd          F          p p<.05
## 2              set_size    3 429 993.80568 1.160813e-192 *
## 3      target_presence    1 143 807.77887 1.077007e-60 *
## 4              task    1 143 103.56653 1.233287e-18 *
## 5      set_size:target_presence    3 429 457.54124 3.036174e-133 *
## 6              set_size:task    3 429 130.08584 6.200153e-60 *
## 7      target_presence:task    1 143 95.28818 1.451633e-17 *
## 8 set_size:target_presence:task    3 429 37.41166 1.705784e-21 *
##      ges
## 2 0.62857261
## 3 0.34080939
## 4 0.03547688
## 5 0.25421847
## 6 0.10306314
## 7 0.01564004
## 8 0.01233006
##
## $`Mauchly's Test for Sphericity`
##              Effect          W          p p<.05
## 2              set_size 0.1805621 2.255039e-50 *
## 5      set_size:target_presence 0.1246345 1.208254e-61 *
## 6              set_size:task 0.2394771 8.327310e-42 *
## 8 set_size:target_presence:task 0.2863555 2.150772e-36 *
##
## $`Sphericity Corrections`
##              Effect          GGe          p[GG] p[GG]<.05          HFe
## 2              set_size 0.5578729 7.577823e-109 * 0.5637096
## 5      set_size:target_presence 0.5423290 1.197865e-73 * 0.5476909
## 6              set_size:task 0.5674096 3.125745e-35 * 0.5735428
```



```
## 8 set_size:target_presence:task 0.6397745 1.200915e-14 * 0.6482876
##      p[HF] p[HF]<.05
## 2 5.924683e-110 *
## 5 2.397734e-74 *
## 6 1.394190e-35 *
## 8 8.266325e-15 *
```

```
table1 <- list(rtaverages$set_size,rtaverages$task) #find breakdown just of setsize and task - less br
table1 #show means so that one can begin to interpret the data. You'll break down rtaverages in differ
```

```
## [[1]]
##      [1] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##      [25] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##      [49] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##      [73] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##      [97] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [121] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [145] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [169] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [193] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [217] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [241] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [265] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [289] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [313] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [337] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [361] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [385] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [409] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [433] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [457] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [481] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [505] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [529] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [553] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [577] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [601] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [625] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [649] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [673] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [697] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [721] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [745] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [769] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [793] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [817] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [841] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [865] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [889] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [913] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [937] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [961] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##     [985] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##    [1009] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
##    [1033] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
```


[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

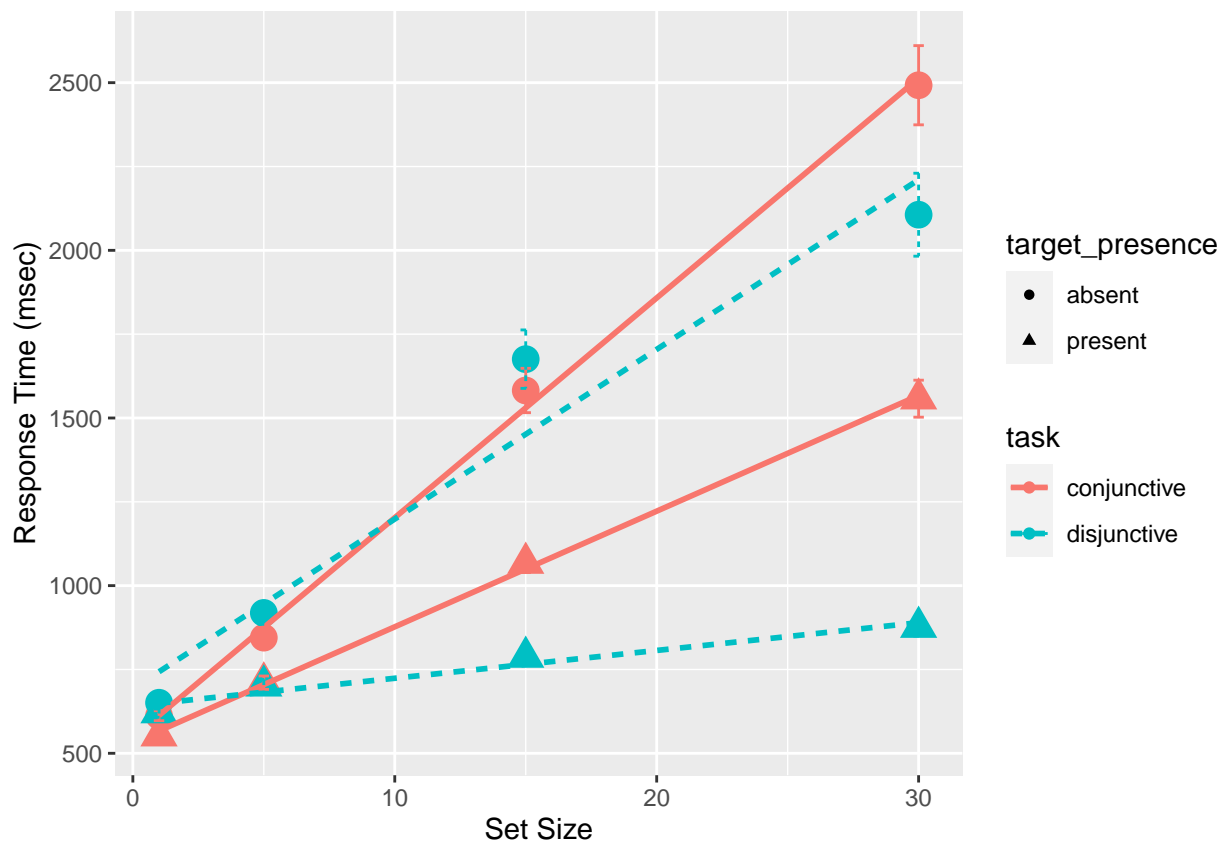
[illegible]

```
## [2263] conjunctive disjunctive conjunctive disjunctive conjunctive disjunctive
## [2269] conjunctive disjunctive conjunctive disjunctive conjunctive disjunctive
## [2275] conjunctive disjunctive conjunctive disjunctive conjunctive disjunctive
## [2281] conjunctive disjunctive conjunctive disjunctive conjunctive disjunctive
## [2287] conjunctive disjunctive conjunctive disjunctive conjunctive disjunctive
## [2293] conjunctive disjunctive conjunctive disjunctive conjunctive disjunctive
## [2299] conjunctive disjunctive conjunctive disjunctive conjunctive disjunctive
## Levels: conjunctive disjunctive
```

```
# The next bit of code is to reproduce Treisman and Gelade's Figure 1, including best lines of fit
rtaverages$set_size_num<-sizes[rtaverages$set_size] # added a new column to rtaverage data frame which
# You need to fill in the XXXs in the ggplot below. Think about what you want to appear on Y axes, and
ggplot(rtaverages,aes(x=set_size_num,y=rt,shape=target_presence,color=task, linetype=task))+ stat_summary
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

```
## `geom_smooth()` using formula 'y ~ x'
```



```
#stat_summary makes a summary of the points, first showing data as points, then adding line of best fit
ca<-lm(rt~set_size_num,data=filter(rtaverages,task=="conjunctive" & target_presence=="absent")) # if you
# cp<-#Deleted piece of code to determine the slope and intercept of the best fitting line for the conj
cp <- lm(rt~set_size_num,data=filter(rtaverages,task=="conjunctive" & target_presence=="present"))
```

```
byseveral <- group_by(goodData,subject_id,target_presence,set_size,task)#Should be just like the analog
pcaverages<-summarize(byseveral,percentCorrect=sum(correct==1)/n()) #gives average accuracy broken down
```

```
## `summarise()` has grouped output by 'subject_id', 'target_presence',
## 'set_size'. You can override using the `.groups` argument.
```

```
model <- ezANOVA(data=byseveral,dv=rt,within=c(set_size,target_presence,task),wid=subject_id) #conduct
```

```
## Warning: Collapsing data to cell means. *IF* the requested effects are a subset
## of the full design, you must use the "within_full" argument, else results may be
## inaccurate.
```

```
model # show results of the ANOVA model
```

```
## $ANOVA
##           Effect DFn DFd           F          p p<.05
## 2           set_size      3 429 1023.37053 4.718545e-195 *
## 3       target_presence      1 143  758.76353 4.762601e-59 *
## 4           task          1 143  118.81909 1.625435e-20 *
## 5  set_size:target_presence      3 429  424.81652 4.991384e-128 *
## 6      set_size:task          3 429  146.33061 2.644614e-65 *
## 7  target_presence:task          1 143  125.14903 2.905726e-21 *
## 8 set_size:target_presence:task      3 429   47.87046 1.053651e-26 *
##
## ges
## 2 0.63650320
## 3 0.32704209
## 4 0.04056876
## 5 0.23773476
## 6 0.11390641
## 7 0.02034845
## 8 0.01572533
##
## $`Mauchly's Test for Sphericity`
##           Effect          W          p p<.05
## 2           set_size 0.1803983 2.116392e-50 *
## 5  set_size:target_presence 0.1194348 6.096605e-63 *
## 6           set_size:task 0.2321395 9.492383e-43 *
## 8 set_size:target_presence:task 0.2838950 1.179251e-36 *
##
## $`Sphericity Corrections`
##           Effect          GGe          p[GG] p[GG]<.05          HFe
## 2           set_size 0.5569078 5.374592e-110 * 0.5627146
## 5  set_size:target_presence 0.5400024 1.591316e-70 * 0.5452941
## 6           set_size:task 0.5646910 4.043705e-38 * 0.5707392
## 8 set_size:target_presence:task 0.6330863 7.575542e-18 * 0.6413698
##
##           p[HF] p[HF]<.05
## 2 4.123231e-111 *
## 5 3.466127e-71 *
## 6 1.693054e-38 *
## 8 4.776154e-18 *
```

```
table1 <- list(pcoverages$set_size, pcoverages$task, pcoverages$percentCorrect) #apply mean function to
table1 # show means so that one can begin to interpret the data
```

```
## [[1]]
## [1] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
## [25] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
## [49] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
## [73] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
## [97] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
## [121] 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30 1 1 5 5 15 15 30 30
```

##	[145]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[169]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[193]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[217]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[241]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[265]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[289]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[313]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[337]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[361]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[385]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[409]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[433]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[457]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[481]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[505]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[529]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[553]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[577]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[601]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[625]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[649]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[673]	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30	1	1	5	5	15	15	30	30
##	[697]	1	1	5	5	15	15	30	30	1	1	5	5												

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```

## [43] 1.0000000 0.9230769 0.9545455 0.9545455 0.3636364 0.9677419 1.0000000
## [50] 0.9600000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [57] 0.9629630 0.9600000 0.9130435 0.9444444 0.9310345 0.9677419 0.6666667
## [64] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9655172 0.9655172
## [71] 1.0000000 1.0000000 1.0000000 0.9600000 0.9629630 0.8965517 0.9523810
## [78] 1.0000000 0.9615385 0.9200000 1.0000000 1.0000000 0.9354839 1.0000000
## [85] 0.9615385 0.9615385 0.9600000 1.0000000 0.9600000 0.9565217 0.8421053
## [92] 1.0000000 0.7916667 1.0000000 0.6800000 0.8709677 0.9666667 1.0000000
## [99] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [106] 0.9666667 1.0000000 1.0000000 0.9600000 1.0000000 0.8148148 1.0000000
## [113] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [120] 1.0000000 0.9600000 0.9629630 1.0000000 1.0000000 0.9130435 1.0000000
## [127] 0.8181818 1.0000000 1.0000000 0.9677419 0.9666667 1.0000000 1.0000000
## [134] 1.0000000 1.0000000 1.0000000 1.0000000 0.8947368 0.9500000 0.9473684
## [141] 0.9473684 1.0000000 0.9333333 1.0000000 1.0000000 0.9687500 1.0000000
## [148] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9444444
## [155] 0.9583333 0.9629630 0.9615385 1.0000000 0.7826087 1.0000000 0.9629630
## [162] 1.0000000 1.0000000 0.9500000 0.9629630 1.0000000 1.0000000 1.0000000
## [169] 1.0000000 0.9000000 0.9629630 0.9333333 0.8260870 1.0000000 0.7083333
## [176] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [183] 1.0000000 1.0000000 1.0000000 0.9000000 0.9565217 0.9565217 1.0000000
## [190] 1.0000000 0.9705882 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [197] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9642857 1.0000000
## [204] 1.0000000 0.8965517 1.0000000 0.9545455 1.0000000 1.0000000 1.0000000
## [211] 0.9545455 1.0000000 0.9565217 1.0000000 0.9600000 1.0000000 1.0000000
## [218] 0.9642857 1.0000000 1.0000000 1.0000000 0.9565217 0.9200000 0.9583333
## [225] 0.9642857 1.0000000 0.9629630 1.0000000 0.9642857 1.0000000 0.9230769
## [232] 1.0000000 1.0000000 0.9047619 1.0000000 0.9310345 1.0000000 0.9523810
## [239] 1.0000000 0.9615385 0.9677419 0.9259259 1.0000000 1.0000000 0.9411765
## [246] 1.0000000 1.0000000 1.0000000 1.0000000 0.9565217 1.0000000 1.0000000
## [253] 0.9393939 0.9500000 0.9090909 1.0000000 0.8214286 0.9200000 1.0000000
## [260] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9090909 0.8800000
## [267] 0.9166667 0.9333333 0.9285714 0.8333333 0.7777778 0.9259259 1.0000000
## [274] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9583333 1.0000000
## [281] 1.0000000 1.0000000 1.0000000 1.0000000 0.9655172 1.0000000 0.9615385
## [288] 0.9600000 0.9600000 0.9130435 1.0000000 1.0000000 1.0000000 1.0000000
## [295] 1.0000000 1.0000000 1.0000000 1.0000000 0.9545455 1.0000000 0.9687500
## [302] 1.0000000 0.8620690 0.9523810 0.9333333 0.7857143 0.9565217 0.9642857
## [309] 0.9230769 0.9523810 0.9310345 1.0000000 0.9500000 0.8181818 0.8888889
## [316] 0.7727273 0.7083333 0.8965517 0.4285714 0.9090909 1.0000000 1.0000000
## [323] 0.9600000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [330] 0.9642857 0.9600000 0.9000000 0.9615385 1.0000000 1.0000000 1.0000000
## [337] 1.0000000 0.9230769 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [344] 1.0000000 1.0000000 0.9583333 0.9285714 0.9259259 0.8260870 1.0000000
## [351] 0.7894737 0.9565217 0.9565217 1.0000000 1.0000000 1.0000000 1.0000000
## [358] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [365] 0.9310345 0.9677419 0.8095238 1.0000000 0.9666667 1.0000000 1.0000000
## [372] 1.0000000 1.0000000 1.0000000 0.9666667 1.0000000 1.0000000 1.0000000
## [379] 1.0000000 1.0000000 0.9032258 1.0000000 0.9500000 0.9545455 0.9677419
## [386] 1.0000000 0.9310345 1.0000000 0.9642857 1.0000000 0.9333333 1.0000000
## [393] 0.8947368 0.8709677 0.9047619 0.9523810 1.0000000 1.0000000 0.8500000
## [400] 0.7142857 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9655172
## [407] 1.0000000 1.0000000 1.0000000 1.0000000 0.9545455 0.9583333 1.0000000
## [414] 1.0000000 0.8846154 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000

```

```

## [421] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [428] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9705882 1.0000000
## [435] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [442] 0.9047619 1.0000000 1.0000000 1.0000000 1.0000000 0.8214286 1.0000000
## [449] 0.9642857 0.9583333 1.0000000 0.9629630 1.0000000 1.0000000 1.0000000
## [456] 1.0000000 1.0000000 0.9615385 0.9200000 1.0000000 0.8800000 0.9310345
## [463] 0.8214286 0.9047619 1.0000000 1.0000000 1.0000000 1.0000000 0.9375000
## [470] 0.9677419 0.9473684 1.0000000 1.0000000 0.9545455 1.0000000 0.9615385
## [477] 0.9705882 0.9473684 0.9032258 0.9545455 0.9565217 0.9230769 1.0000000
## [484] 0.9600000 0.9629630 1.0000000 1.0000000 0.9600000 0.9629630 0.8750000
## [491] 0.9230769 0.7600000 0.9130435 0.9629630 0.9200000 0.9200000 0.9473684
## [498] 1.0000000 0.9600000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [505] 1.0000000 0.9583333 1.0000000 1.0000000 1.0000000 1.0000000 0.9090909
## [512] 1.0000000 0.9565217 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [519] 1.0000000 1.0000000 1.0000000 0.9615385 1.0000000 0.9545455 1.0000000
## [526] 1.0000000 0.9565217 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [533] 1.0000000 1.0000000 1.0000000 1.0000000 0.9523810 0.9200000 1.0000000
## [540] 0.9047619 0.9090909 0.8947368 0.8750000 0.9583333 1.0000000 1.0000000
## [547] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [554] 0.9642857 1.0000000 0.9285714 0.8800000 1.0000000 0.8214286 1.0000000
## [561] 0.9615385 0.9545455 1.0000000 0.9615385 0.9583333 1.0000000 0.9545455
## [568] 1.0000000 1.0000000 0.9285714 0.9545455 0.9166667 0.9230769 0.9615385
## [575] 0.7857143 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [582] 1.0000000 0.9473684 0.9655172 1.0000000 1.0000000 1.0000000 1.0000000
## [589] 1.0000000 1.0000000 0.9032258 1.0000000 0.8571429 1.0000000 1.0000000
## [596] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [603] 1.0000000 0.9600000 0.8800000 0.7600000 0.6500000 1.0000000 0.9629630
## [610] 0.8461538 1.0000000 0.9615385 1.0000000 1.0000000 1.0000000 0.9615385
## [617] 1.0000000 0.8750000 1.0000000 1.0000000 0.8750000 0.9655172 0.8846154
## [624] 1.0000000 0.9615385 1.0000000 1.0000000 1.0000000 1.0000000 0.9523810
## [631] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9090909 0.9130435
## [638] 1.0000000 0.9117647 0.9565217 1.0000000 0.9583333 1.0000000 1.0000000
## [645] 0.9259259 1.0000000 1.0000000 0.9666667 1.0000000 0.9615385 1.0000000
## [652] 1.0000000 0.9130435 0.9200000 0.8518519 1.0000000 1.0000000 0.9629630
## [659] 1.0000000 1.0000000 1.0000000 1.0000000 0.9666667 1.0000000 1.0000000
## [666] 0.9565217 0.9523810 0.9259259 0.9473684 1.0000000 0.9500000 0.9642857
## [673] 0.9642857 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [680] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [687] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [694] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9655172
## [701] 1.0000000 1.0000000 0.9166667 1.0000000 1.0000000 0.9629630 1.0000000
## [708] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [715] 1.0000000 1.0000000 1.0000000 0.9310345 0.9629630 1.0000000 0.9615385
## [722] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9600000 1.0000000
## [729] 1.0000000 0.9000000 0.9047619 0.9444444 0.9642857 1.0000000 0.9600000
## [736] 1.0000000 1.0000000 0.9500000 1.0000000 1.0000000 1.0000000 1.0000000
## [743] 1.0000000 1.0000000 1.0000000 0.9666667 1.0000000 1.0000000 1.0000000
## [750] 1.0000000 0.8620690 1.0000000 1.0000000 1.0000000 0.8571429 1.0000000
## [757] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [764] 0.8571429 0.9642857 0.9500000 1.0000000 1.0000000 0.9166667 0.9600000
## [771] 0.9642857 1.0000000 0.9333333 1.0000000 0.9677419 0.9629630 1.0000000
## [778] 0.8800000 0.8636364 0.8750000 0.9714286 0.9600000 0.8421053 1.0000000
## [785] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [792] 1.0000000 1.0000000 0.9333333 1.0000000 0.9523810 0.9600000 1.0000000

```

```

## [799] 0.7037037 1.0000000 0.8823529 0.9565217 1.0000000 1.0000000 0.9629630
## [806] 1.0000000 0.9629630 1.0000000 0.9696970 0.9629630 0.9615385 0.9600000
## [813] 0.7826087 0.9259259 0.8260870 1.0000000 1.0000000 1.0000000 1.0000000
## [820] 1.0000000 0.9629630 1.0000000 0.9047619 1.0000000 1.0000000 0.9565217
## [827] 1.0000000 0.9615385 0.8695652 0.9642857 0.9310345 1.0000000 0.7692308
## [834] 0.9285714 1.0000000 0.9545455 1.0000000 1.0000000 0.8620690 0.9600000
## [841] 0.9166667 0.7727273 0.9677419 0.8571429 0.8965517 1.0000000 0.8095238
## [848] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9642857 0.9615385
## [855] 1.0000000 1.0000000 1.0000000 0.9259259 0.9375000 1.0000000 1.0000000
## [862] 1.0000000 0.7083333 0.9642857 0.9130435 1.0000000 0.9629630 0.9565217
## [869] 0.9629630 1.0000000 0.9583333 1.0000000 1.0000000 1.0000000 0.9565217
## [876] 1.0000000 1.0000000 1.0000000 1.0000000 0.9411765 0.9333333 1.0000000
## [883] 1.0000000 1.0000000 0.9666667 1.0000000 0.9629630 1.0000000 0.9500000
## [890] 0.8965517 0.8571429 0.8695652 0.9500000 0.9565217 0.8695652 0.8965517
## [897] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [904] 1.0000000 1.0000000 1.0000000 0.9600000 1.0000000 0.8571429 1.0000000
## [911] 0.8148148 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [918] 1.0000000 1.0000000 1.0000000 1.0000000 0.9259259 0.9545455 0.9583333
## [925] 0.9600000 0.9600000 0.8181818 1.0000000 1.0000000 1.0000000 1.0000000
## [932] 1.0000000 0.9583333 1.0000000 1.0000000 1.0000000 0.9600000 1.0000000
## [939] 0.9583333 0.9629630 0.9230769 1.0000000 0.6538462 1.0000000 1.0000000
## [946] 1.0000000 0.9600000 1.0000000 0.9545455 1.0000000 1.0000000 1.0000000
## [953] 1.0000000 1.0000000 0.9600000 1.0000000 1.0000000 1.0000000 0.9310345
## [960] 0.9629630 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [967] 1.0000000 1.0000000 0.9545455 1.0000000 0.9545455 1.0000000 0.9130435
## [974] 1.0000000 0.9200000 0.9655172 0.8695652 1.0000000 1.0000000 1.0000000
## [981] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.8518519 0.9166667
## [988] 0.9428571 0.8000000 0.9600000 0.7500000 1.0000000 1.0000000 1.0000000
## [995] 1.0000000 0.9583333 1.0000000 1.0000000 1.0000000 1.0000000 0.9545455
## [1002] 0.9600000 1.0000000 1.0000000 0.9500000 1.0000000 1.0000000 1.0000000
## [1009] 1.0000000 1.0000000 0.9615385 1.0000000 1.0000000 1.0000000 1.0000000
## [1016] 1.0000000 0.9583333 1.0000000 1.0000000 0.9642857 0.9677419 1.0000000
## [1023] 0.8400000 0.9583333 1.0000000 1.0000000 0.9615385 1.0000000 1.0000000
## [1030] 1.0000000 1.0000000 1.0000000 1.0000000 0.9473684 0.9583333 0.9166667
## [1037] 0.9666667 1.0000000 0.8000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1044] 1.0000000 0.9666667 0.9310345 1.0000000 1.0000000 1.0000000 0.9444444
## [1051] 0.9642857 1.0000000 0.9500000 0.9523810 0.8400000 1.0000000 0.9629630
## [1058] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9583333
## [1065] 0.9565217 1.0000000 1.0000000 0.9600000 0.9655172 0.9473684 0.7000000
## [1072] 1.0000000 1.0000000 0.9677419 1.0000000 1.0000000 1.0000000 1.0000000
## [1079] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1086] 0.9230769 1.0000000 1.0000000 0.8636364 1.0000000 1.0000000 1.0000000
## [1093] 1.0000000 1.0000000 1.0000000 1.0000000 0.9642857 0.7916667 1.0000000
## [1100] 0.9666667 0.9259259 1.0000000 0.9629630 0.9230769 0.9629630 1.0000000
## [1107] 0.6153846 0.9473684 1.0000000 0.9545455 0.9565217 0.9687500 1.0000000
## [1114] 0.9230769 0.6666667 0.9677419 0.6923077 0.9642857 0.5925926 0.8333333
## [1121] 1.0000000 1.0000000 1.0000000 0.9615385 1.0000000 1.0000000 0.9655172
## [1128] 1.0000000 0.9565217 0.9166667 1.0000000 0.9583333 1.0000000 0.9583333
## [1135] 0.9523810 1.0000000 0.9642857 1.0000000 1.0000000 1.0000000 1.0000000
## [1142] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9200000 1.0000000
## [1149] 0.9545455 0.9583333 0.8947368 1.0000000 0.9090909 0.9000000 1.0000000
## [1156] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1163] 1.0000000 0.9677419 1.0000000 0.9565217 0.8333333 1.0000000 0.9523810
## [1170] 1.0000000 1.0000000 0.9230769 0.9600000 1.0000000 1.0000000 0.9615385

```

```

## [1177] 0.9655172 0.9473684 0.9285714 0.9166667 1.0000000 0.9565217 1.0000000
## [1184] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1191] 1.0000000 1.0000000 1.0000000 0.9615385 1.0000000 1.0000000 1.0000000
## [1198] 1.0000000 0.9615385 1.0000000 0.9545455 1.0000000 0.9393939 1.0000000
## [1205] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9583333 0.9411765
## [1212] 0.8965517 0.9032258 0.9230769 0.8695652 1.0000000 0.9565217 1.0000000
## [1219] 1.0000000 1.0000000 0.9545455 1.0000000 1.0000000 1.0000000 1.0000000
## [1226] 0.9565217 1.0000000 0.9000000 0.9285714 0.9642857 0.9047619 1.0000000
## [1233] 0.9629630 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1240] 1.0000000 1.0000000 0.8095238 1.0000000 0.9600000 1.0000000 1.0000000
## [1247] 0.8181818 0.9565217 0.9545455 0.9200000 1.0000000 0.9666667 1.0000000
## [1254] 1.0000000 0.9230769 1.0000000 1.0000000 0.9200000 0.9523810 1.0000000
## [1261] 0.9130435 1.0000000 0.7500000 0.9583333 1.0000000 0.9629630 1.0000000
## [1268] 1.0000000 1.0000000 1.0000000 0.9230769 1.0000000 1.0000000 0.9130435
## [1275] 0.9310345 0.9666667 1.0000000 0.9285714 0.9583333 0.9583333 0.9090909
## [1282] 1.0000000 1.0000000 0.9615385 1.0000000 1.0000000 1.0000000 1.0000000
## [1289] 0.9642857 0.9200000 1.0000000 1.0000000 0.9259259 1.0000000 0.8620690
## [1296] 1.0000000 0.9615385 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1303] 1.0000000 1.0000000 1.0000000 0.9047619 0.9615385 1.0000000 0.9615385
## [1310] 1.0000000 0.9565217 0.9600000 0.9500000 0.9629630 1.0000000 1.0000000
## [1317] 0.9629630 1.0000000 1.0000000 1.0000000 1.0000000 0.9565217 1.0000000
## [1324] 0.9615385 0.9130435 0.9259259 0.9047619 0.9565217 0.9565217 1.0000000
## [1331] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9629630
## [1338] 0.9545455 1.0000000 1.0000000 0.9523810 1.0000000 0.8275862 0.9523810
## [1345] 0.9130435 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1352] 1.0000000 1.0000000 1.0000000 0.9615385 1.0000000 0.9629630 1.0000000
## [1359] 0.9642857 1.0000000 0.9523810 1.0000000 1.0000000 1.0000000 1.0000000
## [1366] 1.0000000 1.0000000 1.0000000 1.0000000 0.9615385 1.0000000 0.9545455
## [1373] 1.0000000 1.0000000 0.7826087 0.9500000 1.0000000 1.0000000 1.0000000
## [1380] 0.9310345 0.9565217 1.0000000 0.9200000 1.0000000 0.8695652 0.8888889
## [1387] 0.9130435 1.0000000 0.9629630 1.0000000 0.8800000 0.9500000 0.9047619
## [1394] 0.9655172 1.0000000 1.0000000 0.9666667 1.0000000 1.0000000 1.0000000
## [1401] 1.0000000 0.9523810 1.0000000 1.0000000 1.0000000 0.9629630 0.8750000
## [1408] 0.8333333 0.9500000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1415] 1.0000000 0.9583333 1.0000000 0.9142857 0.9333333 0.9629630 0.9642857
## [1422] 0.9285714 0.9428571 0.9615385 0.9642857 1.0000000 1.0000000 1.0000000
## [1429] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1436] 1.0000000 1.0000000 1.0000000 0.5714286 1.0000000 0.9545455 0.9696970
## [1443] 1.0000000 1.0000000 1.0000000 0.9259259 1.0000000 1.0000000 1.0000000
## [1450] 0.8823529 1.0000000 0.9666667 1.0000000 1.0000000 0.8636364 1.0000000
## [1457] 0.9600000 1.0000000 0.9500000 1.0000000 1.0000000 1.0000000 1.0000000
## [1464] 1.0000000 0.9600000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1471] 0.9714286 1.0000000 0.9615385 0.9565217 1.0000000 1.0000000 0.9600000
## [1478] 1.0000000 1.0000000 1.0000000 1.0000000 0.9259259 0.9200000 1.0000000
## [1485] 1.0000000 1.0000000 0.9375000 1.0000000 0.9600000 1.0000000 0.9677419
## [1492] 1.0000000 0.9565217 0.9545455 1.0000000 1.0000000 1.0000000 1.0000000
## [1499] 1.0000000 0.9473684 0.9629630 1.0000000 0.9583333 0.9565217 0.9677419
## [1506] 1.0000000 1.0000000 1.0000000 0.9655172 1.0000000 0.9615385 1.0000000
## [1513] 1.0000000 0.8750000 0.8928571 0.9615385 0.9523810 0.9615385 0.9166667
## [1520] 0.9130435 1.0000000 0.9642857 1.0000000 0.9642857 1.0000000 1.0000000
## [1527] 1.0000000 1.0000000 0.9600000 1.0000000 0.9545455 1.0000000 0.9285714
## [1534] 0.9600000 0.9523810 1.0000000 0.9615385 1.0000000 0.9285714 1.0000000
## [1541] 1.0000000 1.0000000 1.0000000 0.9600000 0.9583333 1.0000000 0.9545455
## [1548] 0.9523810 0.8518519 0.9642857 0.6956522 0.9600000 1.0000000 1.0000000

```



```

## [1555] 0.9687500 1.0000000 1.0000000 0.9500000 1.0000000 1.0000000 1.0000000
## [1562] 0.9200000 1.0000000 0.9230769 1.0000000 1.0000000 0.9130435 0.9666667
## [1569] 0.9166667 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9655172
## [1576] 1.0000000 1.0000000 0.9600000 0.9615385 0.9583333 0.9666667 0.9615385
## [1583] 0.9523810 0.9677419 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1590] 1.0000000 1.0000000 1.0000000 1.0000000 0.9615385 1.0000000 0.9230769
## [1597] 0.9642857 1.0000000 0.9259259 1.0000000 0.9523810 0.9166667 1.0000000
## [1604] 1.0000000 1.0000000 0.9333333 1.0000000 1.0000000 0.9310345 0.8076923
## [1611] 0.9259259 0.9642857 0.8461538 0.9500000 0.9285714 0.9259259 1.0000000
## [1618] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1625] 1.0000000 1.0000000 0.8800000 0.9642857 0.8750000 0.9285714 0.7857143
## [1632] 0.9310345 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1639] 1.0000000 1.0000000 1.0000000 0.9047619 1.0000000 1.0000000 1.0000000
## [1646] 1.0000000 1.0000000 1.0000000 0.9500000 0.9629630 1.0000000 1.0000000
## [1653] 1.0000000 1.0000000 0.9629630 1.0000000 1.0000000 0.8695652 1.0000000
## [1660] 1.0000000 0.8500000 0.8400000 0.5652174 0.9090909 1.0000000 1.0000000
## [1667] 1.0000000 1.0000000 0.9583333 1.0000000 0.9583333 1.0000000 1.0000000
## [1674] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9230769 1.0000000
## [1681] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9615385 1.0000000
## [1688] 1.0000000 1.0000000 0.8965517 1.0000000 0.9642857 0.9600000 0.9583333
## [1695] 0.7857143 1.0000000 0.9500000 0.9523810 1.0000000 1.0000000 1.0000000
## [1702] 0.9615385 0.9629630 0.8333333 1.0000000 0.9310345 1.0000000 0.9655172
## [1709] 0.9565217 0.9583333 0.9565217 1.0000000 0.9545455 1.0000000 1.0000000
## [1716] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9285714 0.9310345
## [1723] 0.9166667 1.0000000 0.7600000 1.0000000 0.4285714 1.0000000 1.0000000
## [1730] 1.0000000 1.0000000 0.9523810 1.0000000 1.0000000 0.9600000 1.0000000
## [1737] 0.9523810 1.0000000 1.0000000 0.9655172 0.9565217 1.0000000 0.9200000
## [1744] 0.9500000 1.0000000 1.0000000 0.9545455 1.0000000 1.0000000 1.0000000
## [1751] 1.0000000 1.0000000 1.0000000 0.9545455 1.0000000 1.0000000 1.0000000
## [1758] 1.0000000 0.8846154 1.0000000 0.9615385 0.9285714 0.8928571 0.9259259
## [1765] 1.0000000 1.0000000 0.9444444 0.9473684 1.0000000 0.9545455 1.0000000
## [1772] 0.9130435 0.8750000 0.9565217 0.9062500 0.9677419 1.0000000 1.0000000
## [1779] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1786] 0.9545455 1.0000000 1.0000000 0.9565217 1.0000000 0.9642857 1.0000000
## [1793] 0.7619048 0.9629630 1.0000000 0.9565217 1.0000000 0.8823529 1.0000000
## [1800] 1.0000000 0.8620690 0.8695652 0.8709677 1.0000000 0.8275862 1.0000000
## [1807] 0.2500000 0.9629630 0.9629630 1.0000000 1.0000000 0.9259259 1.0000000
## [1814] 1.0000000 1.0000000 1.0000000 1.0000000 0.9565217 1.0000000 1.0000000
## [1821] 0.8846154 0.9677419 0.8620690 0.9600000 1.0000000 1.0000000 0.9687500
## [1828] 1.0000000 0.9545455 0.9615385 0.9130435 1.0000000 1.0000000 0.9090909
## [1835] 1.0000000 0.9411765 0.9285714 0.9583333 0.8148148 0.9677419 1.0000000
## [1842] 1.0000000 0.9666667 0.9600000 1.0000000 1.0000000 1.0000000 1.0000000
## [1849] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9545455 0.9230769
## [1856] 0.9583333 0.9200000 1.0000000 0.9655172 1.0000000 1.0000000 0.9666667
## [1863] 1.0000000 1.0000000 0.9200000 0.9666667 0.8571429 0.9000000 0.8260870
## [1870] 0.9500000 0.4400000 0.8888889 1.0000000 0.9583333 1.0000000 1.0000000
## [1877] 1.0000000 1.0000000 1.0000000 1.0000000 0.9655172 0.9615385 0.9523810
## [1884] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9666667
## [1891] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9615385 1.0000000
## [1898] 1.0000000 0.9677419 1.0000000 0.9310345 1.0000000 0.8400000 1.0000000
## [1905] 0.9565217 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1912] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9523810 1.0000000
## [1919] 0.8461538 0.9687500 1.0000000 0.9615385 1.0000000 1.0000000 1.0000000
## [1926] 0.9473684 1.0000000 1.0000000 1.0000000 1.0000000 0.8148148 1.0000000

```

```

## [1933] 0.7083333 0.9354839 0.8461538 1.0000000 0.9629630 0.9615385 0.9629630
## [1940] 1.0000000 1.0000000 1.0000000 0.9629630 0.9523810 1.0000000 0.9166667
## [1947] 0.9565217 0.9259259 0.9130435 0.9166667 0.8695652 0.9655172 0.8421053
## [1954] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [1961] 0.9677419 0.9655172 0.9583333 0.8750000 0.9230769 1.0000000 0.7391304
## [1968] 1.0000000 1.0000000 0.9629630 1.0000000 0.9677419 1.0000000 1.0000000
## [1975] 1.0000000 1.0000000 1.0000000 0.9565217 0.9677419 0.9473684 1.0000000
## [1982] 0.9642857 0.8260870 0.9090909 1.0000000 1.0000000 1.0000000 1.0000000
## [1989] 0.9655172 1.0000000 1.0000000 1.0000000 1.0000000 0.9500000 1.0000000
## [1996] 0.9629630 0.9523810 0.9565217 0.9090909 0.9666667 0.9629630 1.0000000
## [2003] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9565217
## [2010] 0.9411765 0.9583333 0.9200000 0.8518519 1.0000000 0.7931034 0.9642857
## [2017] 0.9473684 1.0000000 1.0000000 0.9565217 1.0000000 0.9600000 0.9629630
## [2024] 1.0000000 1.0000000 0.8500000 0.9583333 0.9259259 0.8750000 1.0000000
## [2031] 0.8260870 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2038] 1.0000000 0.9600000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2045] 0.9565217 1.0000000 0.8800000 1.0000000 0.9047619 1.0000000 1.0000000
## [2052] 0.9583333 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9583333
## [2059] 1.0000000 0.9230769 0.9565217 0.9545455 0.6071429 1.0000000 0.9166667
## [2066] 0.9615385 0.9565217 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2073] 0.9615385 1.0000000 1.0000000 0.9687500 1.0000000 1.0000000 0.9583333
## [2080] 1.0000000 1.0000000 0.4800000 1.0000000 0.9565217 0.9583333 0.8636364
## [2087] 0.9259259 0.9655172 1.0000000 0.5600000 1.0000000 0.8888889 0.8461538
## [2094] 0.8571429 0.6956522 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2101] 0.9565217 1.0000000 1.0000000 1.0000000 1.0000000 0.9333333 1.0000000
## [2108] 0.9642857 1.0000000 0.9642857 0.9166667 1.0000000 0.9166667 1.0000000
## [2115] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2122] 1.0000000 0.9166667 0.9600000 0.9545455 1.0000000 0.8888889 0.9583333
## [2129] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2136] 1.0000000 1.0000000 0.9642857 0.9629630 0.9600000 1.0000000 1.0000000
## [2143] 0.9333333 0.9642857 1.0000000 1.0000000 0.8947368 1.0000000 1.0000000
## [2150] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.8695652
## [2157] 0.9545455 0.9565217 0.9047619 1.0000000 0.9565217 0.9333333 1.0000000
## [2164] 0.9629630 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2171] 0.9583333 0.9565217 0.8846154 1.0000000 0.6923077 1.0000000 0.9565217
## [2178] 0.9600000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9583333
## [2185] 1.0000000 0.9600000 0.9545455 1.0000000 0.9259259 1.0000000 0.9523810
## [2192] 0.9615385 0.9583333 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2199] 1.0000000 1.0000000 1.0000000 0.9583333 0.9642857 0.9642857 1.0000000
## [2206] 1.0000000 0.9629630 0.9600000 1.0000000 1.0000000 0.9565217 1.0000000
## [2213] 1.0000000 0.9473684 0.9545455 1.0000000 1.0000000 0.9666667 1.0000000
## [2220] 1.0000000 0.9565217 0.9354839 1.0000000 1.0000000 1.0000000 1.0000000
## [2227] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2234] 1.0000000 1.0000000 0.9655172 1.0000000 1.0000000 1.0000000 1.0000000
## [2241] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2248] 1.0000000 1.0000000 0.9655172 1.0000000 0.9545455 0.7600000 1.0000000
## [2255] 0.7142857 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2262] 1.0000000 1.0000000 1.0000000 1.0000000 0.9583333 1.0000000 1.0000000
## [2269] 0.8750000 0.9500000 0.8666667 1.0000000 0.9310345 1.0000000 0.9523810
## [2276] 1.0000000 1.0000000 1.0000000 0.9523810 1.0000000 1.0000000 0.8571429
## [2283] 0.9310345 0.8421053 0.9047619 1.0000000 0.8965517 0.9655172 0.9166667
## [2290] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [2297] 0.9615385 0.9655172 0.9615385 0.9642857 0.9200000 1.0000000 1.0000000
## [2304] 1.0000000

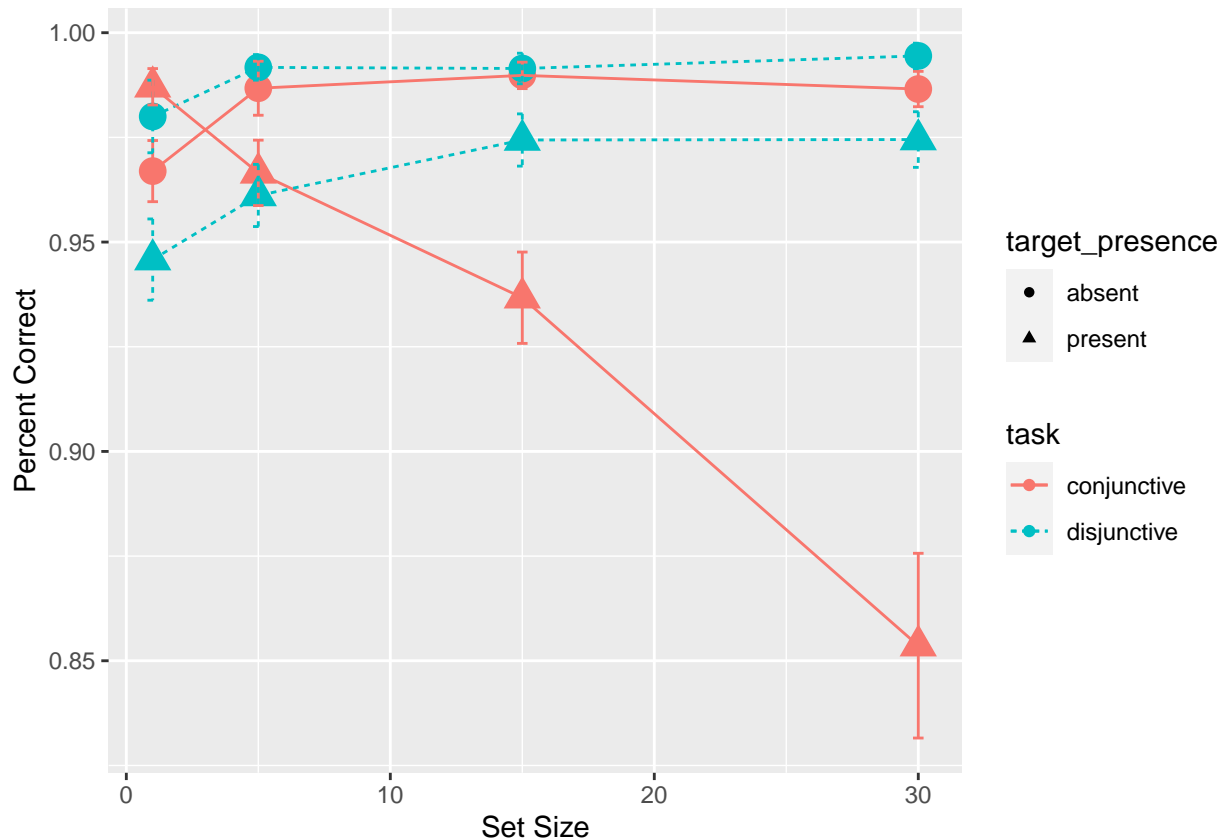
```

```
pcaverages$set_size_num<-sizes[pcaverages$set_size] # added a new column to rtaverage data frame which
# Plot the results (which turn out to be surprising) that mirror the response times results, but for ac
head(pcaverages)
```

```
## # A tibble: 6 x 6
## # Groups:   subject_id, target_presence, set_size [3]
##   subject_id target_presence set_size task      percentCorrect set_size_num
##   <fct>      <fct>          <fct> <fct>          <dbl>          <dbl>
## 1 2000081652 absent            1   conjunctive      1            1
## 2 2000081652 absent            1   disjunctive      1            1
## 3 2000081652 absent            5   conjunctive      1            5
## 4 2000081652 absent            5   disjunctive      1            5
## 5 2000081652 absent           15   conjunctive      1           15
## 6 2000081652 absent           15   disjunctive  0.941          15
```

```
ggplot(pcaverages,aes(x=set_size_num,y=percentCorrect,shape=target_presence,color=task, linetype=task)).
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

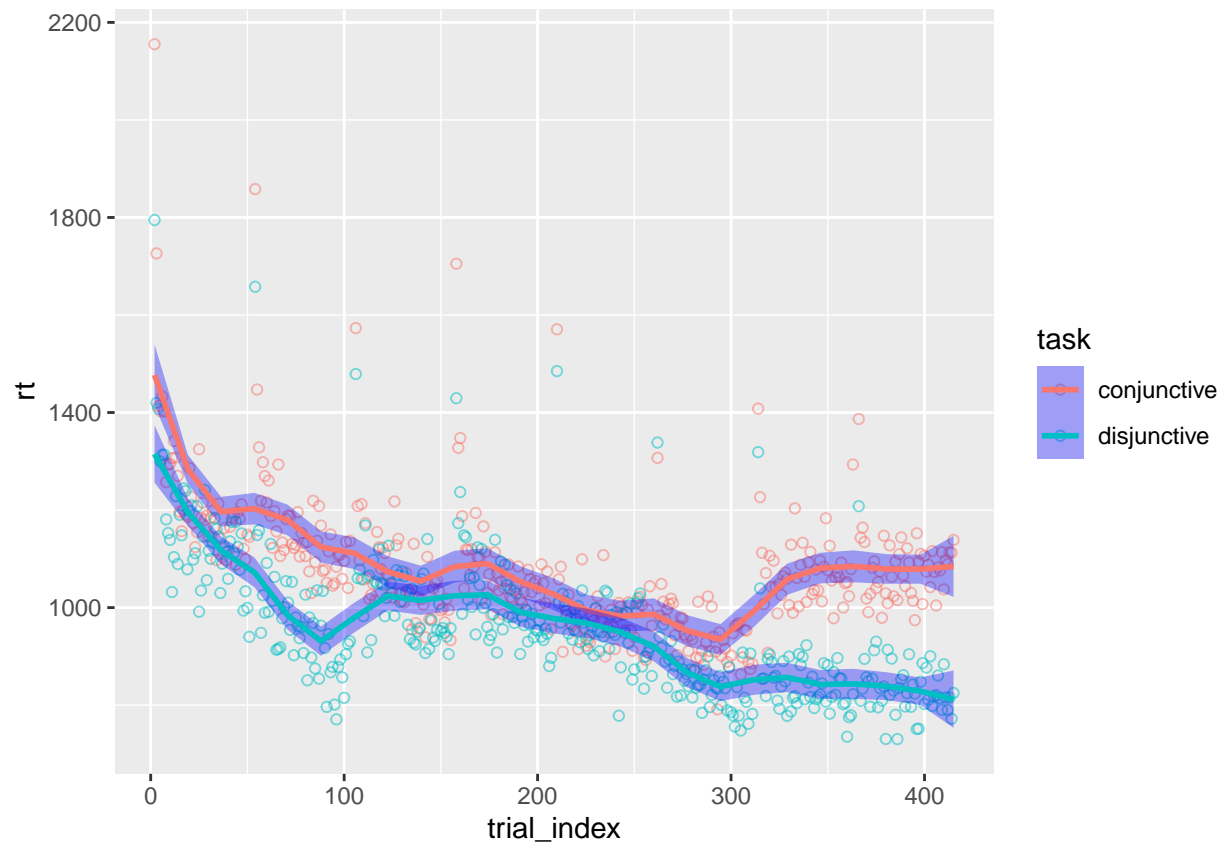


```
#The following analysis begins to look at learning effects. Are there improvements over all trials with
byseveral<-group_by(goodSearchTimes,task,trial_index) #break data down by task and which trial number.
averages<-summarize(byseveral,rt=mean(rt,trim=.1)) #show averages for each subject for each cell of the
```

```
## `summarise()` has grouped output by 'task'. You can override using the
## `.groups` argument.
```

```
ggplot(averages,aes(x=trial_index,y=rt,colour=task))+geom_point(shape=1,alpha=1/2)+ geom_smooth(method=
```

```
## `geom_smooth()` using formula 'y ~ x'
```



```
#Are there improvements over trials within each block?
```

```
goodSearchTimes$trialInBlock<-(goodSearchTimes$trial_index-1) %% 52 # add another column to describe wh  
byseveral<-group_by(goodSearchTimes,subject_id,task,trialInBlock) #break data down by subject, schedule  
averages<-summarize(byseveral,rt=mean(rt,trim=.3)) #show averages for each subject for each cell of the
```

```
## `summarise()` has grouped output by 'subject_id', 'task'. You can override  
## using the `.groups` argument.
```

```
ggplot(averages,aes(x=trialInBlock,y=rt,colour=task))+geom_point(shape=1,alpha=1/2)+ geom_smooth(method=
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 13 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 13 rows containing missing values (geom_point).
```

