Appendix 1

```
### Model code for simulation 1

## create contingency task trials

create_trials <- function(A, B, C, D, vector_length, shuffle=FALSE){

  event_frequencies <- rep(c(1,2,3,4), times = c(A,B,C,D))

  event_pairs <- matrix(c(rep(c(1,1,1), each = vector_length),
                          rep(c(1,0,1), each = vector_length),
                          rep(c(0,1,1), each = vector_length),
                          rep(c(0,0,1), each = vector_length)),
                      ncol= vector_length*3,
                      byrow=TRUE)

  event_matrix <- event_pairs[event_frequencies,]

  if(shuffle==FALSE) {return(event_matrix)
    }else{
    return(event_matrix[sample(1:length(event_frequencies)),])
    }

}

## Run the model function

run_model <- function(trials,probe){
  memory <- trials

#probe memory with cue only

  similarities <- c()
  for(i in 1:dim(memory)[1]){
    similarities[i] <- lsa::cosine(probe,memory[i,])
  }
  similarities[is.nan(similarities)]<-0

  # generate echo
  weighted_memory <- memory*(similarities^3)
  echo <- colSums(weighted_memory)

  return(echo)
}

### DEFINE conditions

conditions <- list(list(outcome_density = .2,
                        delta_p = 0,
                        frequencies = c(6,24,6,24)),
                   list(outcome_density = .8,
                        delta_p = 0,
                        frequencies = c(24,6,24,6)),
                   list(outcome_density = .33,
                        delta_p = .467,
                        frequencies = c(17,13,3,27)),
                   list(outcome_density = .67,
                        delta_p = .467,
                        frequencies = c(27,3,13,17))
                   )

model_data<-data.frame() #initialize data frame

## Run the model across conditions

for(i in 1:4){
  trials <- create_trials(A=conditions[[i]]$frequencies[1],
                          B=conditions[[i]]$frequencies[2],
                          C=conditions[[i]]$frequencies[3],
                          D=conditions[[i]]$frequencies[4],
                          vector_length = 10,
                          shuffle=TRUE)
  model_output <- run_model(trials=trials,probe = rep(c(1,0,1),each=10))
  sim_data <- data.frame(outcome_density = conditions[[i]]$outcome_density,
                         delta_p = conditions[[i]]$delta_p,
                         expectation = model_output[11])
  model_data <- rbind(model_data,sim_data)
}

# print table
#knitr::kable(model_data)
```

```
# plot
library(ggplot2)

ggplot(model_data,aes(x=outcome_density,
                      y=expectation,
                      group=delta_p,
                      color=delta_p))+
  geom_line()+
  geom_point()
```