# Appendix: R code

# K : number of rivals

# d : disadvantage (in standard deviations)

# n : number of simulations

n <- 100000

K <- 1:15

D <- seq(0,3, by=0.1)

Output <- expand.grid(k=K, d=D, prob=NA)

# The simulation itself

for(d in D){ # for each of a range of levels of disadvantage

for (k in K){ # for between 1 and 15 rivals

A.win <- vector("numeric", n)

for (i in 1:n){

B <- rnorm(k) # k draws from a normal distribution with mean of 0 and sd of 1

A <- rnorm(1, -d) # one draw from a normal distribution with a mean of -d and sd of 1

A.win[i] <- as.numeric(A > max(B)) # produces a vector with 1 if A is the greatest number; 0 otherwise

}

prob.A.win <- sum(A.win) / n # calculates the proportion of times where A is the greatest value

Output[Output$k==k & Output$d==d, "prob"] <- prob.A.win

}

}

# Code for producing the graphs

contourplot(prob ~ k \* d, data=Output, cuts=20)

save(Output, file="SimulationOutput.rData")

?contourplot

require(lattice)

png("fig1\_kis1.png", height=800, width=800)

plot(prob ~ d, data=subset(Output, k==1),

type="l", ylab="Probability of getting a job per attempt",

xlab="Fitness disadvantage (Standard deviations)",

main="Relationship between relative fitness and probability \nof getting a job when competing against one healthy competitor", lwd=2, ylim=c(0, 1))

dev.off()

require(lattice)

png("attempts.png", height=800, width=800)

plot((1/prob) ~ d, data=subset(Output, k==1),

type="l", ylab="Expected number of attempts",

xlab="Fitness disadvantage (Standard deviations)",

main="Relationship between relative fitness, number of competitors, \nand probability of getting a job", log="y", lwd=2, ylim=c(1, 200))

lines((1/prob) ~ d, data=subset(Output, k==2), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==3), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==4), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==5), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==6), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==7), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==8), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==8), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==9), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==10), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==11), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==12), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==13), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==14), lty="dashed")

lines((1/prob) ~ d, data=subset(Output, k==15), lwd=2, lty="dashed")

legend("bottomright",

legend=c("One competitor", "intermediate numbers (two to 14 competitors)", "15 competitors"),

lwd=c(2,1,2),

lty=c("solid", "dashed", "dashed"))

dev.off()

#####

png("probs.png", height=800, width=800)

plot(prob ~ d, data=subset(Output, k==1),

type="l", ylab="Probability of getting a job on any one occasion",

xlab="Fitness disadvantage (Standard deviations)",

main="Relationship between relative fitness, number of competitors, \nand probability of getting a job", log="y", lwd=2)

lines(prob ~ d, data=subset(Output, k==2), lty="dashed")

lines(prob ~ d, data=subset(Output, k==3), lty="dashed")

lines(prob ~ d, data=subset(Output, k==4), lty="dashed")

lines(prob ~ d, data=subset(Output, k==5), lty="dashed")

lines(prob ~ d, data=subset(Output, k==6), lty="dashed")

lines(prob ~ d, data=subset(Output, k==7), lty="dashed")

lines(prob ~ d, data=subset(Output, k==8), lty="dashed")

lines(prob ~ d, data=subset(Output, k==8), lty="dashed")

lines(prob ~ d, data=subset(Output, k==9), lty="dashed")

lines(prob ~ d, data=subset(Output, k==10), lty="dashed")

lines(prob ~ d, data=subset(Output, k==11), lty="dashed")

lines(prob ~ d, data=subset(Output, k==12), lty="dashed")

lines(prob ~ d, data=subset(Output, k==13), lty="dashed")

lines(prob ~ d, data=subset(Output, k==14), lty="dashed")

lines(prob ~ d, data=subset(Output, k==15), lwd=2, lty="dashed")

legend("bottomright",

legend=c("One competitor", "intermediate numbers (two to 14 competitors)", "15 competitors"),

lwd=c(2,1,2),

lty=c("solid", "dashed", "dashed"))

dev.off()

####

png("ContourPlot.png", width=800, height=800)

contourplot(1/prob ~ d \* k, data=Output, at=c(2,5,10,25, 50,100,200), xlab="Disadvantage (Standard deviations)", ylab="Number of competitors", main="Contour plot of expected number of attempts required to secure a job")

dev.off()

#####

png("ProbAtFixedDisadvantage.png", width=800, height=800)

Output[Output$d==0.5,] -> tmp ; tmp[order(tmp$prob),]

barplot(tmp$prob, ylab="Probability of getting a job following application", xlab="Number of rivals", names.arg=as.character(1:15))

dev.off()

rm(tmp)