20170605_Mig.Bot_Vary

July 10, 2017

0.1 Effects of migration and bottleneck on the strength and formation of cyanogenesis clines

In this notebook, I look at the effects of migration rate and population bottlenecks on the strength and proportion of significant positive and negative clines in cyanogenesis.

Positive clines refer to increases in the frequency of HCN across space (relative to the starting population).

Negative clines refer to decreases in the frequency of HCN across space (relative to the starting population).

Note that all figures show means or proportion with associated 95% confidence intervals. However, since I ran a separate linear model (Frequency of HCN ~ Distance from starting population) for every generation in every simulation, the number of models over which the CI's are being calculated is very large. This has the evvef of making the CI's so small as to be barely distinguishable in some figures.

```
In [4]: library(plyr)
        library(ggplot2)
        library(data.table, lib="~/Rpackages")
        library(Rmisc, lib = "~/Rpackages")
        library(dplyr)
        library(broom)
```

```
Loading required package: lattice
-------------------------------------------------------------------------------
data.table + dplyr code now lives in dtplyr.
Please library(dtplyr)!
-------------------------------------------------------------------------------

Attaching package: dplyr

The following objects are masked from package:data.table:

    between, first, last

The following objects are masked from package:plyr:

    arrange, count, desc, failwith, id, mutate, rename, summarise,
    summarize

The following objects are masked from package:stats:

    filter, lag

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

```
In [5]: #Working directory for datasets varying migration rate and bottleneck proportion
        setwd('/scratch/research/projects/trifolium/SEC_Simulation.Evolutionary.Clines/SEC_Data/
        Drift.Migration/1D/Mig_Bot_Vary')

        #Load datasets that will be used for analyses
        Bot_SlopeSum <- fread('20170706_Bot_SlopeSum.csv', header = T)
        MigRate_SlopeSum <- fread('20170706_MigRate_SlopeSum.csv', header = T)
        Bot_SlopeSum_Gen <- fread('20170706_Bot_SlopeSum_Gen.csv', header = T)
        MigRate_SlopeSum_Gen <- fread('20170706_MigRate_SlopeSum_Gen.csv', header = T)
```

```
In [6]: #Themes used for plotting
        ng1=theme(aspect.ratio=0.7,panel.background = element_blank(),
                  panel.grid.major = element_blank(),
                  panel.grid.minor = element_blank(),
                  panel.border=element_blank(),
                  axis.line.x = element_line(color="black",size=1),
                  axis.line.y = element_line(color="black",size=1),
                  axis.ticks=element_line(color="black"),
                  axis.text=element_text(color="black",size=15),
                  axis.title=element_text(color="black",size=1),
                  axis.title.y=element_text(vjust=2,face="bold",size=15),
                  axis.title.x=element_text(vjust=0.1,face="bold",size=15),
                  axis.text.x=element_text(size=13),
                  axis.text.y=element_text(size=13),
                  legend.position = "right", legend.direction="vertical",
                  legend.text=element_text(size=11), legend.key = element_rect(fill = "white"),
                  legend.title = element_text(size=13,face="bold"),legend.key.size = unit(0.5,
        "cm"))

        ng1.45=theme(aspect.ratio=0.7,panel.background = element_blank(),
                  panel.grid.major = element_blank(),
                  panel.grid.minor = element_blank(),
                  panel.border=element_blank(),
                  axis.line.x = element_line(color="black",size=1),
                  axis.line.y = element_line(color="black",size=1),
                  axis.ticks=element_line(color="black"),
                  axis.text=element_text(color="black",size=15),
                  axis.title=element_text(color="black",size=1),
                  axis.title.y=element_text(vjust=2,face="bold",size=15),
                  axis.title.x=element_text(vjust=0.1,face="bold",size=15),
                  axis.text.x=element_text(size=13,angle=45,hjust=1),
                  axis.text.y=element_text(size=13),
                  legend.position = "right", legend.direction="vertical",
                  legend.text=element_text(size=11), legend.key = element_rect(fill = "white"),
                  legend.title = element_text(size=13,face="bold"),legend.key.size = unit(0.5,
        "cm"))
```

0.1.1 Mean slope by migration rate and bottleneck strength. Averaged across all generations where the matrix is full and across all 1000 simulations

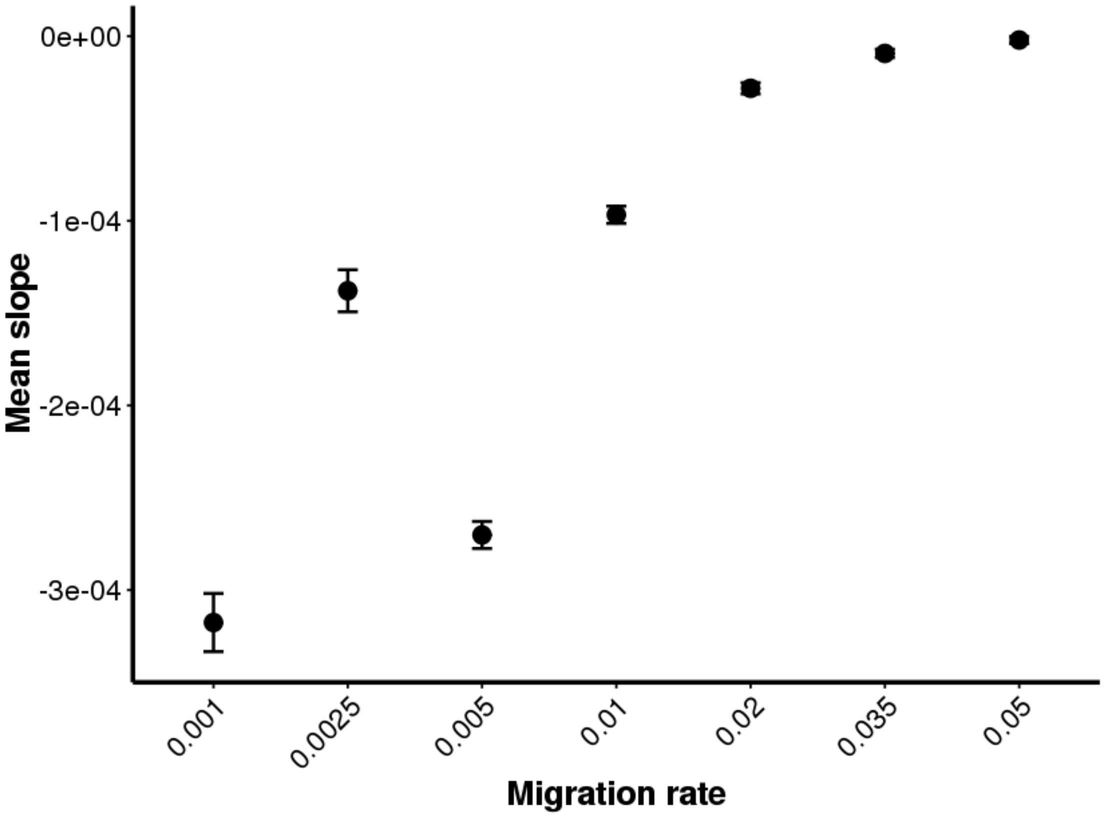
The following two plots show the effect of migration rate and population bottlenecks on the mean slope of clines. Means were calculated by averaging across all 1000 simulations and only include slopes calculated from models when the matrix (50 cells) was filled.

Main results

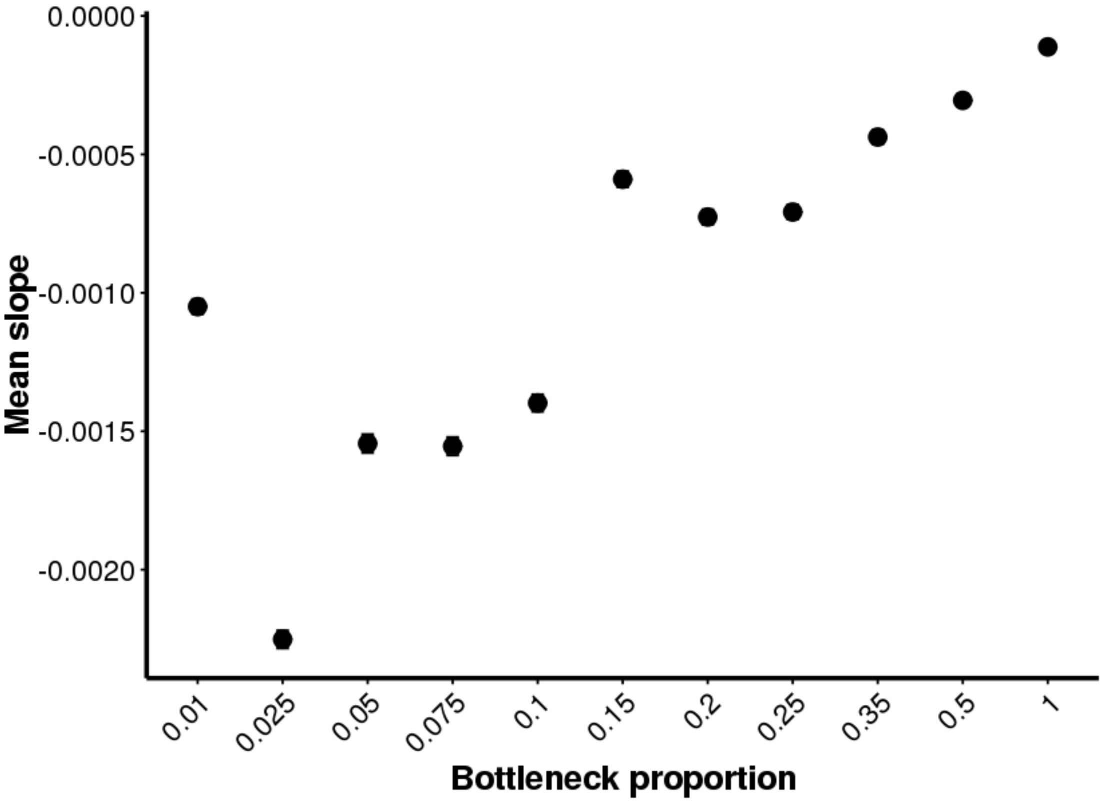
1. Increasing migration rates result in weaker clines (i.e. slopes getting closer to 0). *Notice the weird result at m = 0.0025. Note quite sure what is going on here.*

2. Stronger population bottlenecks (e.g. bottleneck = 0.0025) result in stronger clines. *Notice weird result at bottleneck = 0.001. I suspect I know what is going on here and it relates to the fact that these are average values and the matrix did not fill in many simulations because of the strong reduction in population size under strong bottlenecks. Remember that the probability of crating populations decreaases with decreasing population size.*

```
In [8]: plotMeanSlope_Mig_rate <- ggplot(MigRate_SlopeSum, aes(x = factor(Mig_rate), y = mean))
        +
            geom_errorbar(aes(ymin = mean - ci.lower, ymax = mean + ci.upper), width=0.15,
        size=0.7,
            position = position_dodge(width = 0.5)) +
            ylab("Mean slope") + xlab("Migration rate") + geom_point(size = 3.5, color =
        "black")
        plotMeanSlope_Mig_rate + ng1.45
```



```
In [70]: plotMeanSlope_Bot_rate <- ggplot(Bot_SlopeSum, aes(x = factor(bot), y = mean)) +
             geom_errorbar(aes(ymin = mean - ci.lower, ymax = mean + ci.upper), width=0.15,
         size=0.7,
             position = position_dodge(width = 0.5))+
             ylab("Mean slope") + xlab("Bottleneck proportion") + geom_point(size = 3.5, color =
         "black")
         plotMeanSlope_Bot_rate + ng1.45
```



0.1.2 Proportion of significantly positive and negative clines. Averaged across all generations where the matrix is full and across all 1000 simulations

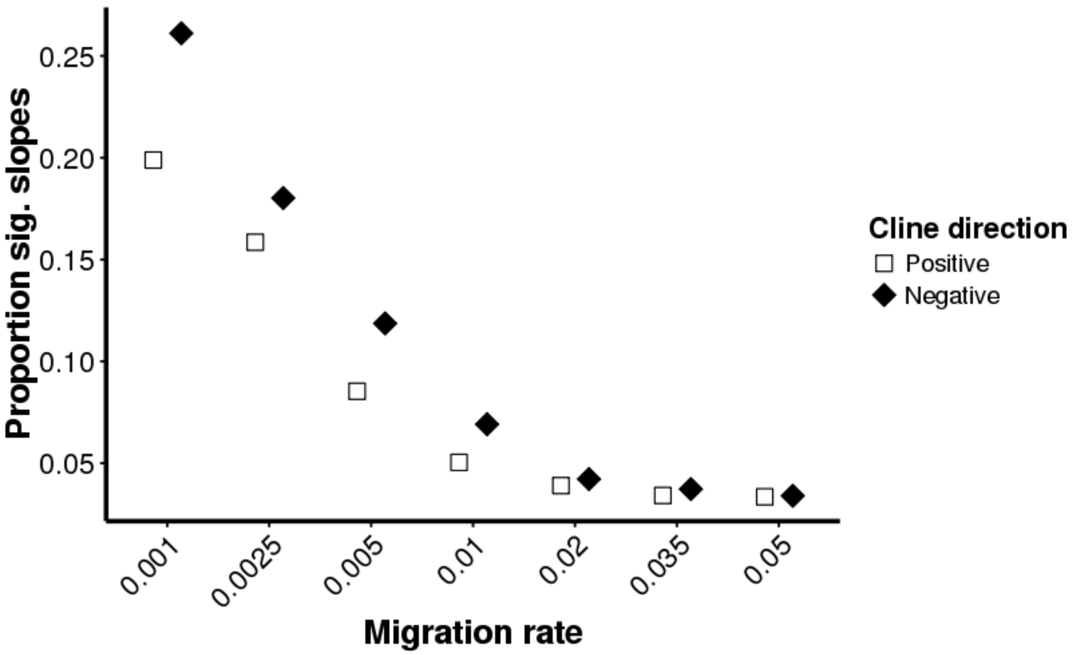
The following two plots show the effect of migration and population bottlenecks on the proportion of significant cyanogenesis clines (both positive and negative). Proportions were calculated across all 1000 simulations and only include clines calculated from models when the matrix (50 cells) was filled.

Main results

1. The proportion of significant clines decreases with increasing migration.
2. For migration rates that do not swamp out clines (i.e. m < 0.02), the proportion of significant negative clines (i.e. loss of HCN across space) is always greater than the proportion of significantly positive clines (i.e. increase in HCN across space).
3. Stronger population bottlenecks are associated with a greater proportion of significant clines. However, this only seems to be true for negative clines; the proportion of significantly positive clines shows little change (a bit of a hump-shaped curve...greatest proportion at intermediate migration rates??).

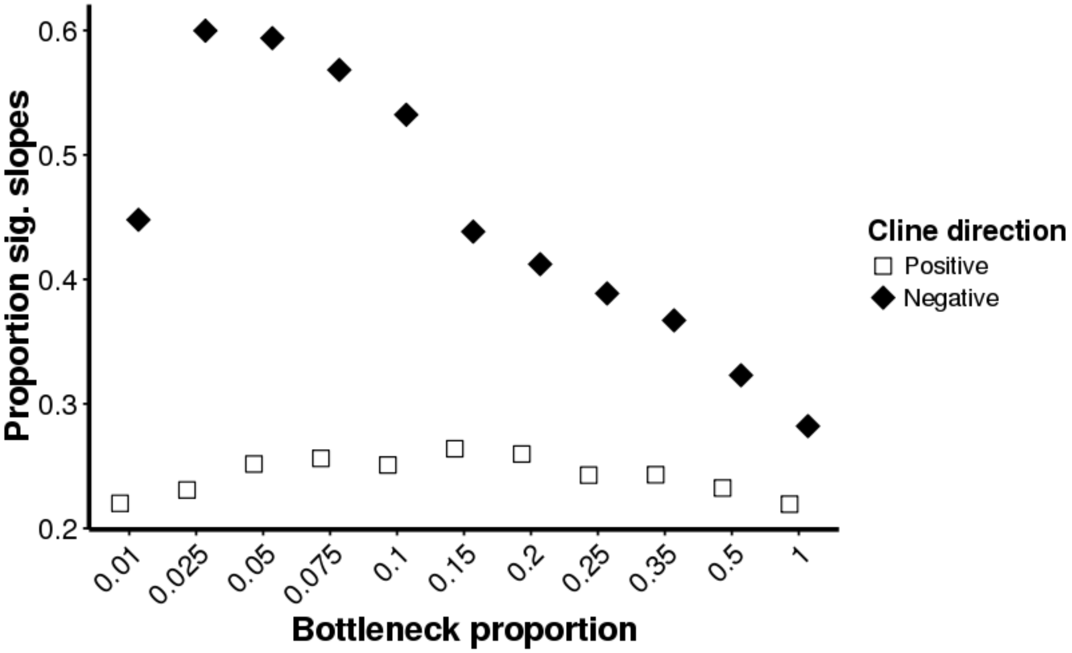
```
In [10]: #Extra columns that will be used to create melting dataset for plotting proportion of
         positive and negative slopes
         MigRate_SlopeSum_Melt <- dplyr::select(MigRate_SlopeSum, Mig_rate, prop_sigPos,
         ci.lower.Pos, ci.upper.Pos, prop_sigNeg,
           ci.lower.Neg, ci.upper.Neg)

         #Melt dataframe
         dm1 <- melt(MigRate_SlopeSum_Melt[,c("Mig_rate", "prop_sigPos", "ci.lower.Pos",
         "ci.upper.Pos")],
                     id=c("Mig_rate", "ci.lower.Pos", "ci.upper.Pos"))
         dm2 <- melt(MigRate_SlopeSum_Melt[,c("Mig_rate", "prop_sigNeg", "ci.lower.Neg",
         "ci.upper.Neg")],
                     id=c("Mig_rate", "ci.lower.Neg", "ci.upper.Neg"))

         #Rename columns
         setnames(dm1, old = c("ci.lower.Pos", "ci.upper.Pos"), new = c("ci.lower", "ci.upper"))
         setnames(dm2, old = c("ci.lower.Neg", "ci.upper.Neg"), new = c("ci.lower", "ci.upper"))

         #Merge melted dataframes
         MigRate_SlopeSum_Melt <- rbind(dm1, dm2)
```

```
In [15]: #Plot proportion of significant slopes by migration rate
         plotPropSlope_Mig_rate <- ggplot(MigRate_SlopeSum_Melt, aes(x = factor(Mig_rate), y =
         value,
             shape = variable, fill = variable)) +
             geom_errorbar(aes(ymin = value - ci.lower, ymax = value + ci.upper), width=0.15,
         size = 0.7,
             position = position_dodge(width = 0.55)) +
             ylab("Proportion sig. slopes") + xlab("Migration rate") + geom_point(size = 3.5,
         color = "black",
             position = position_dodge(width = 0.55)) +
             scale_shape_manual(labels = c("Positive", "Negative"),values=c(22, 23))+
             scale_fill_manual(labels = c("Positive", "Negative"),values=c("white", "black")) +
             scale_y_continuous(breaks = seq(from = 0, to = 0.25, by = 0.05)) +
             labs(shape = 'Cline direction', fill = 'Cline direction')
         plotPropSlope_Mig_rate + ng1.45
```



```
In [19]: #Extrca columns that will be used to create melting dataset for plotting proportion of
         positive and negative slopes
         Bot_SlopeSum_Melt <- dplyr::select(Bot_SlopeSum, bot, prop_sigPos, ci.lower.Pos,
         ci.upper.Pos, prop_sigNeg,
           ci.lower.Neg, ci.upper.Neg)

         #Melt dataframe
         dm1 <- melt(Bot_SlopeSum_Melt[,c("bot", "prop_sigPos", "ci.lower.Pos", "ci.upper.Pos")],
                     id=c("bot", "ci.lower.Pos", "ci.upper.Pos"))
         dm2 <- melt(Bot_SlopeSum_Melt[,c("bot", "prop_sigNeg", "ci.lower.Neg", "ci.upper.Neg")],
                     id=c("bot", "ci.lower.Neg", "ci.upper.Neg"))

         #Rename columns
         setnames(dm1, old = c("ci.lower.Pos", "ci.upper.Pos"), new = c("ci.lower", "ci.upper"))
         setnames(dm2, old = c("ci.lower.Neg", "ci.upper.Neg"), new = c("ci.lower", "ci.upper"))

         #Merge melted dataframes
         Bot_SlopeSum_Melt <- rbind(dm1, dm2)
```

```
In [18]: #Plot proportion of significant slopes by migration rate
         plotPropSlope_bot <- ggplot(Bot_SlopeSum_Melt, aes(x = factor(bot), y = value, shape =
         variable, fill = variable)) +
             geom_errorbar(aes(ymin = value - ci.lower, ymax = value + ci.upper), width=0.15,
         size = 0.7,
             position = position_dodge(width = 0.55)) +
             ylab("Proportion sig. slopes") + xlab("Bottleneck proportion") + geom_point(size =
         3.5, color = "black",
             position = position_dodge(width = 0.55)) +
             scale_shape_manual(labels = c("Positive", "Negative"), values=c(22, 23))+
             scale_fill_manual(labels = c("Positive", "Negative"), values=c("white", "black")) +
             labs(shape = 'Cline direction', fill = 'Cline direction')
         plotPropSlope_bot + ng1.45
```



0.1.3 How does the mean slope and proportion of significant clines vary with increasing number of generations since the matrix has filled? Does this differ by migration rate?

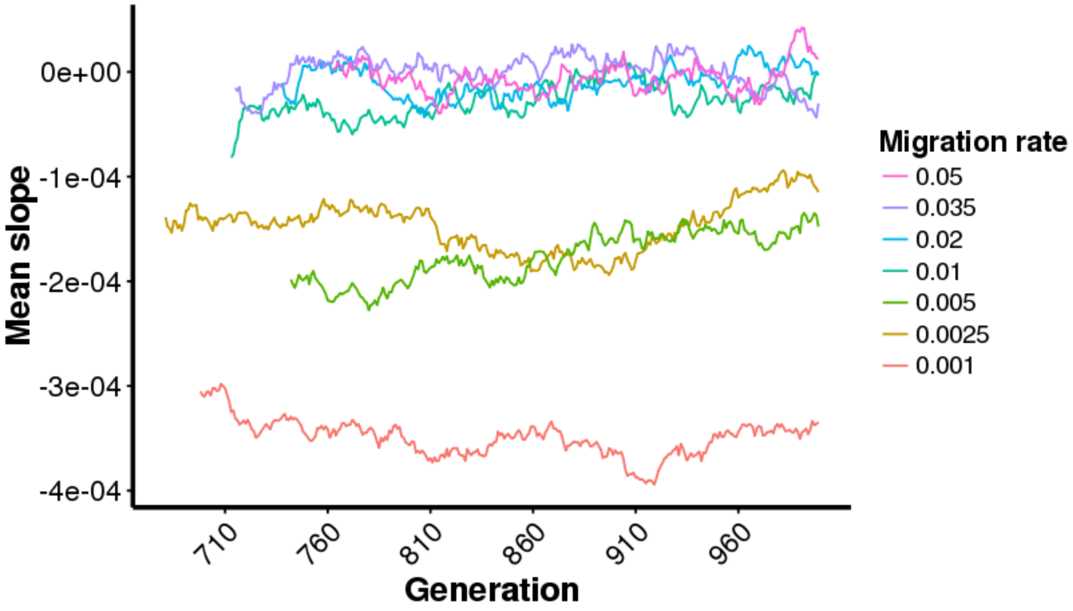
Once the matrix is full, there are no more population bottlenecks and HCN frequencies are governed solely by within-population dynamics and migration. Thus we might expect clines to weaken over time due to the constant mixing of alleles every generation (from migration, if included).

Using only clines for which the matrix was full in all 1000 generations, the following three plots show how the mean slope of clines and the proportion of significant clines changes with increasing number of generations since the matrix filled. Lines are colored by migration rate.

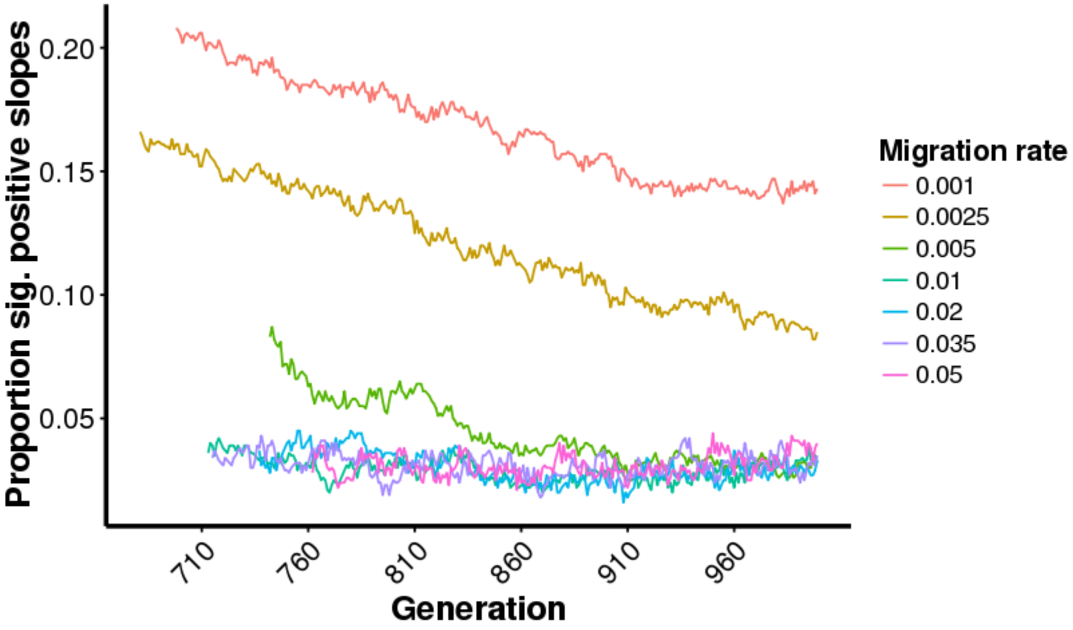
Main results

1. The mean slope of clines shows little change between the generation where the matrix first filled and the final generation of the simulation. This is true regardless of the migration rate.
2. The proportion of significant clines (both positive and negative) decreases with increasing number of generations since the matrix has filled, but only for the three lowest migration rate values. This is consistant with lower migration rates requiring longer to homogenize allele frequencies, resulting in a gradual decrease in the proportion of significant clines over time.
3. For the lowest migration rates, the decrease in the proportion of significant clines with increasing generations has not plateaued, suggesting they may continue to decrease even further if given the chance.

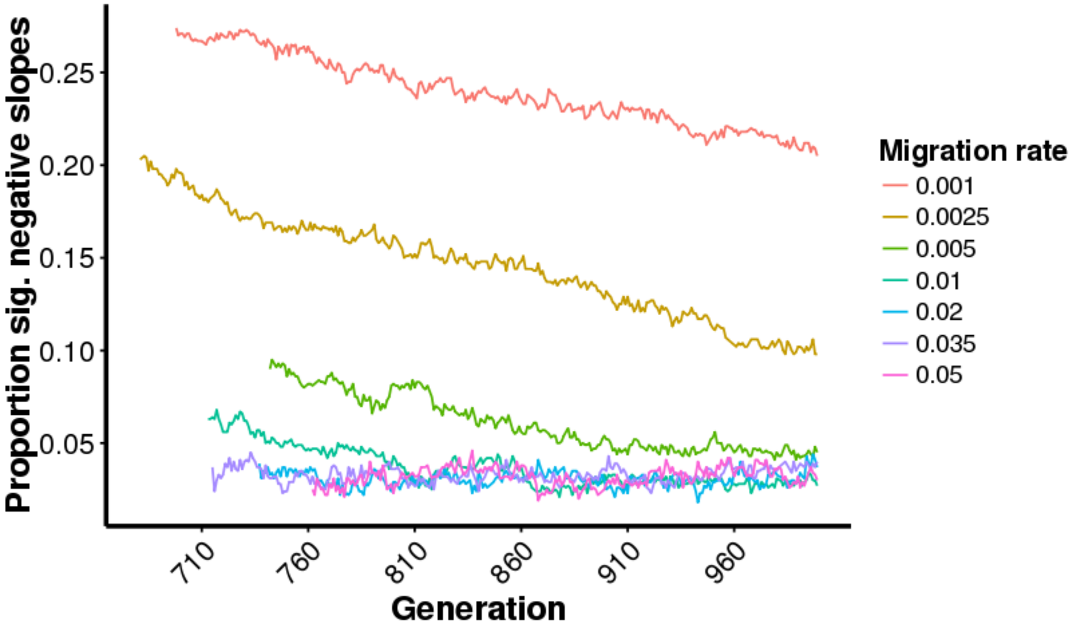
```
In [24]: #Subset to start at the generation where the matrix was filled for all 1000 generations
         MigRate_SlopeSum_Gen_AllFill <- subset(MigRate_SlopeSum_Gen, n == 1000)

         #Plot mean slope by generation number, colored by migration rate
         plotMeanSlope_MigRate_Gen <- ggplot(MigRate_SlopeSum_Gen_AllFill, aes(x = Generation, y
         = mean,
             group = factor(Mig_rate), color = factor(Mig_rate))) +
             ylab("Mean slope") + xlab("Generation") + geom_line(size = 0.5, aes(group =
         factor(Mig_rate),
             color = factor(Mig_rate))) +
             scale_x_continuous(breaks = seq(from = 310, to = 1000, by = 50)) +
             labs(color = 'Migration rate') +
             guides(color = guide_legend(reverse = TRUE))
         plotMeanSlope_MigRate_Gen + ng1.45
```



```
In [26]: plotPropPos_MigRate_Gen <- ggplot(MigRate_SlopeSum_Gen_AllFill, aes(x = Generation, y =
         prop_sigPos,
             group = factor(Mig_rate), color = factor(Mig_rate))) +
             ylab("Proportion sig. positive slopes") + xlab("Generation") + geom_line(size = 0.5,
         aes(group = factor(Mig_rate),
             color = factor(Mig_rate))) +
             scale_x_continuous(breaks = seq(from = 310, to = 1000, by = 50)) +
             labs(color = 'Migration rate')
         plotPropPos_MigRate_Gen + ng1.45
```



```
In [42]: plotPropNeg_MigRate_Gen <- ggplot(MigRate_SlopeSum_Gen_AllFill, aes(x = Generation, y =
         prop_sigNeg,
             group = factor(Mig_rate), color = factor(Mig_rate))) +
             ylab("Proportion sig. negative slopes") + xlab("Generation") + geom_line(size = 0.5,
         aes(group = factor(Mig_rate),
             color = factor(Mig_rate))) +
             scale_x_continuous(breaks = seq(from = 310, to = 1000, by = 50)) +
             scale_y_continuous(breaks = seq(from = 0, to = 0.25, by = 0.05)) +
             labs(color = 'Migration rate')
         plotPropNeg_MigRate_Gen + ng1.45
```



0.1.4 How does the mean slope and proportion of significant clines vary with increasing number of generations since the matrix has filled? Does this differ by bottleneck proportion?

Once the matrix is full, there are no more population bottlenecks and HCN frequencies are governed solely by within-population dynamics (there is no migration when varying the strength of bottlenecks). Given no migration, we would not expect the the clines to disappear following the filling of the matrix. Rather, we might expect that stronger initial bottlenecks produce stronger clines, which are maintained following the filling of the matrix.

Using only clines for which the matrix was full in all 1000 generations, the following three plots show how the mean slope of clines and the proportion of significant clines with increasing number of generations since the matrix filled. Lines are colored by bottleneck strength.

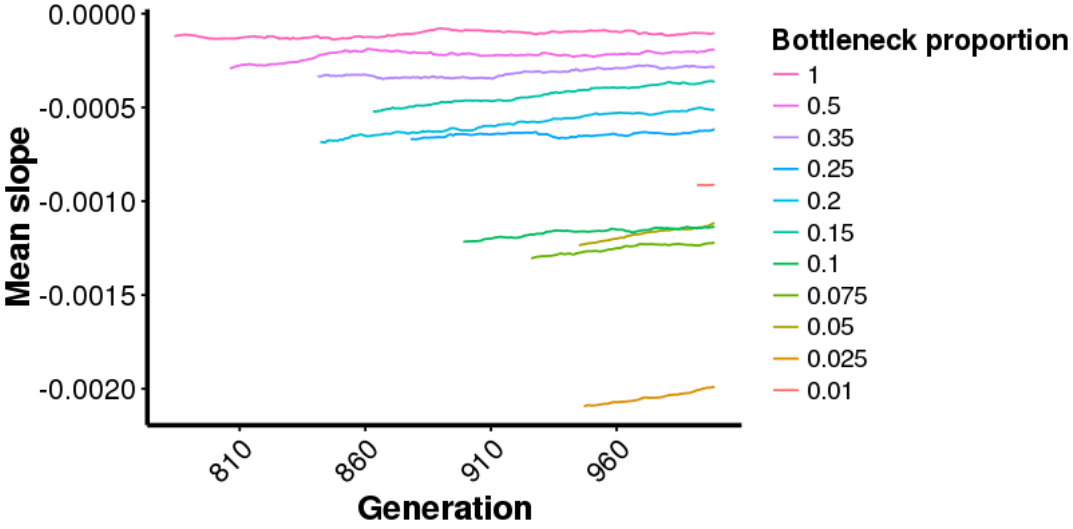
Why are the lines of differing lengths? The lines differ dramatically in length because I am only including generations for which the matrix is filled in all 1000 simulations. The matrix does not fill in the same number of generations every simulation because of stochastic variation having to do with the probability of creating populations. Given that the probability of creating a population is maximized when a population is at carying capacity, and that stronger population bottlenecks (e.g. bottleneck = 0.01) result in greater reductions in population sizes, the matrix fills slower when bottlenecks are stronger. This results in shorter lines for stronger bottleneck values where this stochastic variation is greater.

Main results

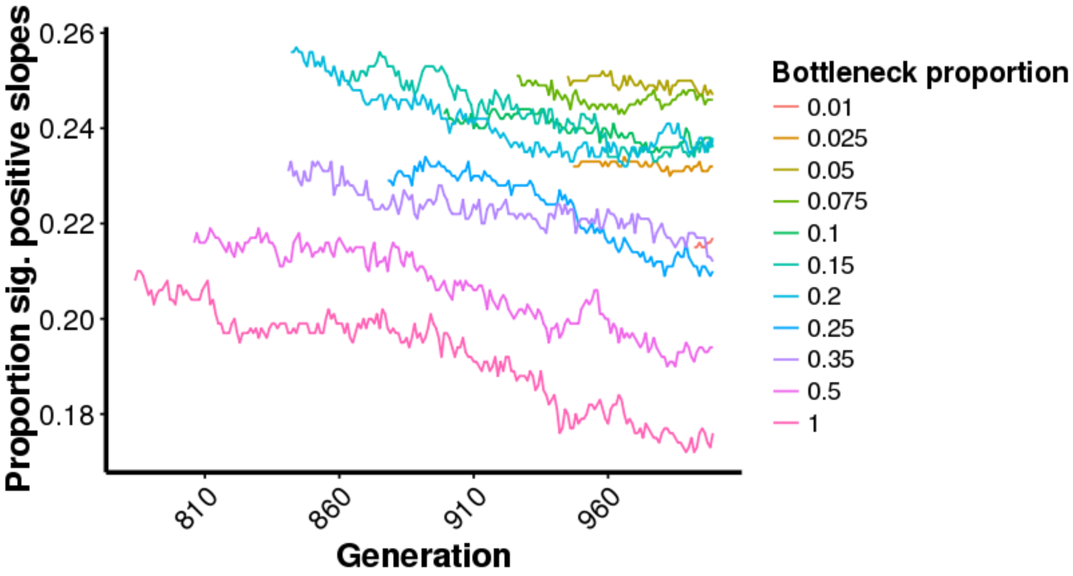
1. There is little change in the mean slope of clines with increasing time since the matrix has been filled. This does not differ according to the strength of population bottlenecks
2. The proportion of significant positive clines (i.e. increases in HCN across space) decreases with increasing time since the matrix has filled. This effect seems strongest when bottlenecks the bottlenecks that generated the clines were at their weakest (e.g. bottleneck = 1.0).

3. There is little change in the proportion of significantly negative clines with increasing time since the matrix filled. This is true for att bottleneck strengths.

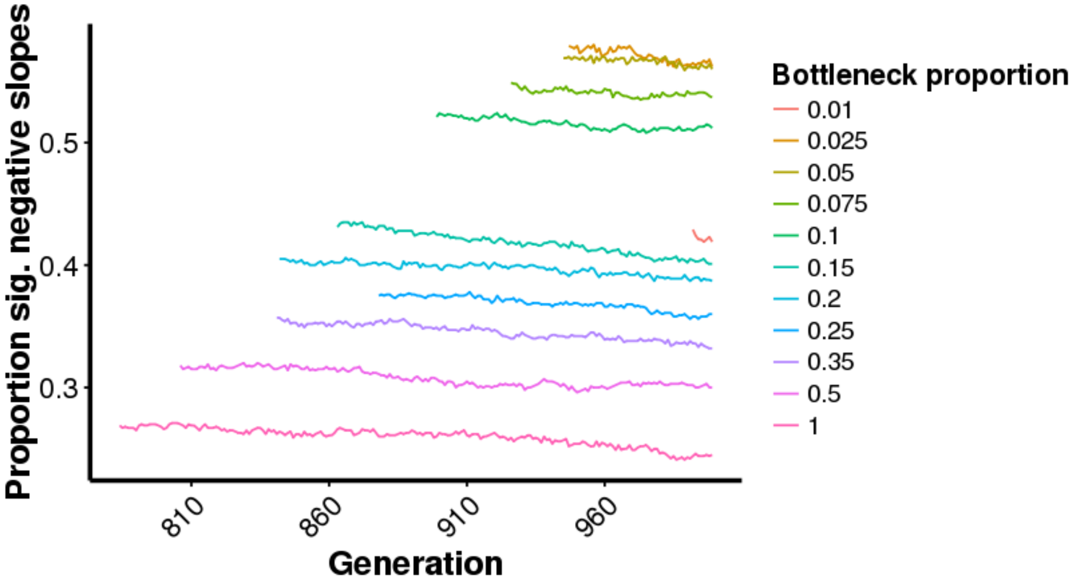
```
In [29]: #Subset data to start at generation where matrix is full in all 1000 generations
         Bot_SlopeSum_Gen_AllFill <- subset(Bot_SlopeSum_Gen, n == 1000)

         #Plot mean slope of clines by generation number, colored by bottleneck proportion
         plotMeanSlope_Bot_Gen <- ggplot(Bot_SlopeSum_Gen_AllFill, aes(x = Generation, y = mean,
             group = factor(bot), color = factor(bot))) +
             ylab("Mean slope") + xlab("Generation") + geom_line(size = 0.5, aes(group =
         factor(bot), color = factor(bot))) +
             scale_x_continuous(breaks = seq(from = 310, to = 1000, by = 50)) +
             labs(color = 'Bottleneck proportion') +
             guides(color = guide_legend(reverse = TRUE))
         plotMeanSlope_Bot_Gen + ng1.45
```



```
In [32]: plotPropPos_Bot_Gen <- ggplot(Bot_SlopeSum_Gen_AllFill, aes(x = Generation, y =
         prop_sigPos, group = factor(bot),
             color = factor(bot))) +
             ylab("Proportion sig. positive slopes") + xlab("Generation") + geom_line(size = 0.5,
         aes(group = factor(bot),
             color = factor(bot))) +
             scale_x_continuous(breaks = seq(from = 310, to = 1000, by = 50)) +
             labs(color = 'Bottleneck proportion')
         plotPropPos_Bot_Gen + ng1.45
```



```
In [33]: plotPropNeg_Bot_Gen <- ggplot(Bot_SlopeSum_Gen_AllFill, aes(x = Generation, y =
         prop_sigNeg, group = factor(bot),
             color = factor(bot))) +
             ylab("Proportion sig. negative slopes") + xlab("Generation") + geom_line(size = 0.5,
         aes(group = factor(bot),
             color = factor(bot))) +
             scale_x_continuous(breaks = seq(from = 310, to = 1000, by = 50)) +
             labs(color = 'Bottleneck proportion')
         plotPropNeg_Bot_Gen + ng1.45
```



0.1.5 Do we still see evidence of an effect of migration and population bottlenecks on the strength and proportion of significant clines when looking only al the last generation?

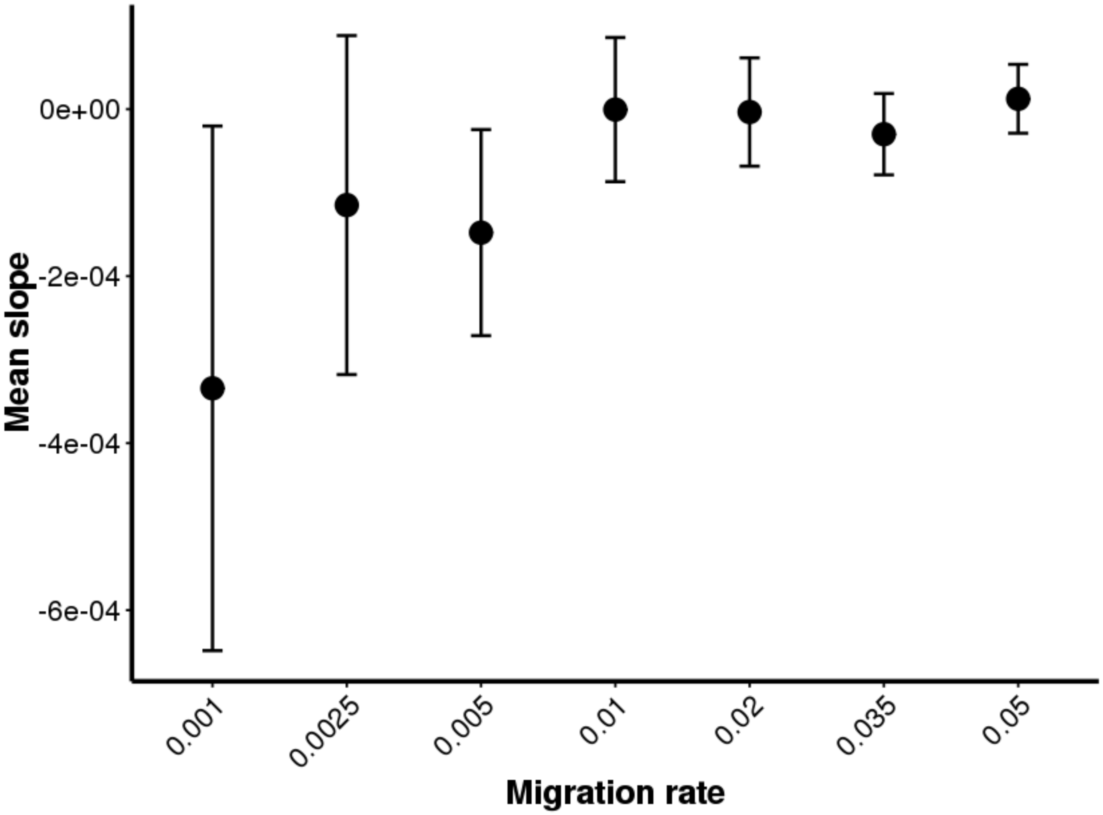
Given that populations have had hundreds of generations to "equilibrate" following the filling of the landscape, we might expect the effects of migration and bottlenecks on the strength and proportion of significant clines to disappear over time (there is some evidence that they do for low migration rates, see above). Thus, the following four plots examine the effects of migration and bottlenecks by subsetting to include only the final generation of all simulations.

Main results

1. Patterns generally consistent with those found above with once exception: The proportion of significant negative clines is only greater than the propotion of significant positive clines

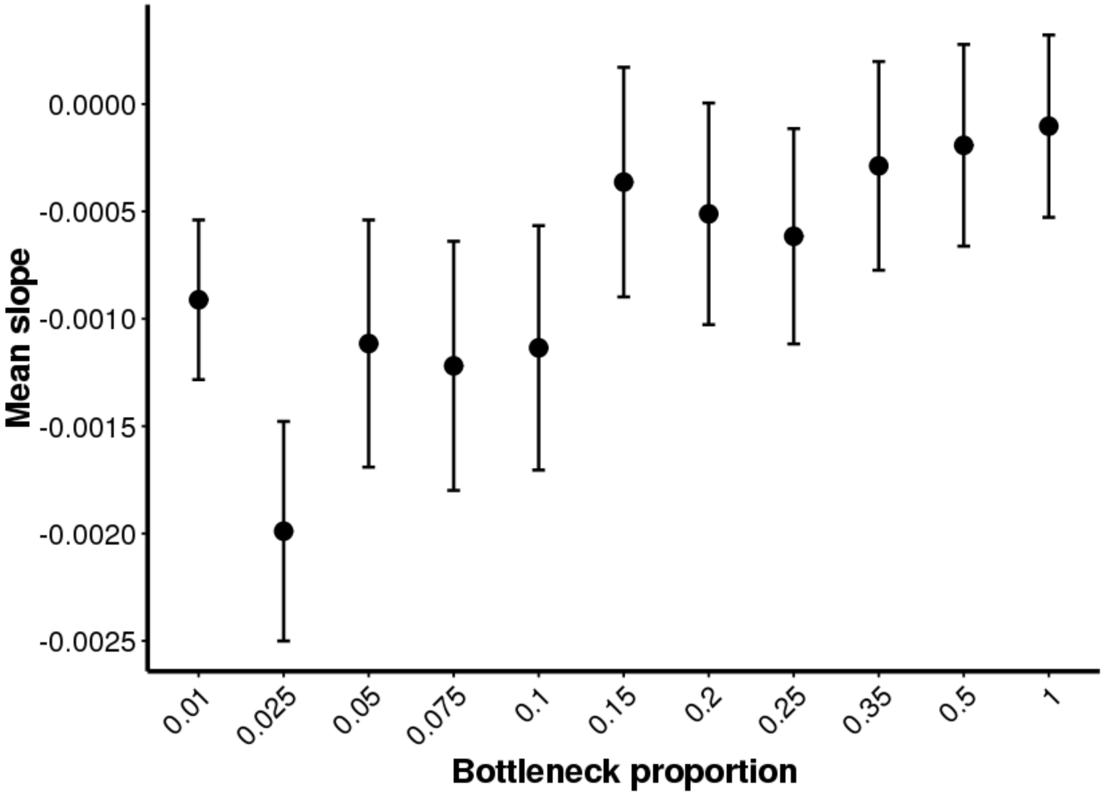
for the lowest migration value. This is consistent with a gradual decrease in the proportion of significant clines with increasing time since the matrix filled (see above).

```
In [34]: #Subset to include only the last generation
         MigRate_SlopeSum_Gen_999 <- subset(MigRate_SlopeSum_Gen, Generation == 999)

         #Plot mean slope by migration rate
         plotMeanSlope_MigRate_999 <- ggplot(MigRate_SlopeSum_Gen_999, aes(x = factor(Mig_rate),
         y = mean)) +
             geom_errorbar(aes(ymin = mean - ci.lower, ymax = mean + ci.upper), width=0.15,
         size=0.7)+
             ylab("Mean slope") + xlab("Migration rate") + geom_point(size = 4.5, color =
         "black")
         plotMeanSlope_MigRate_999 + ng1.45
```



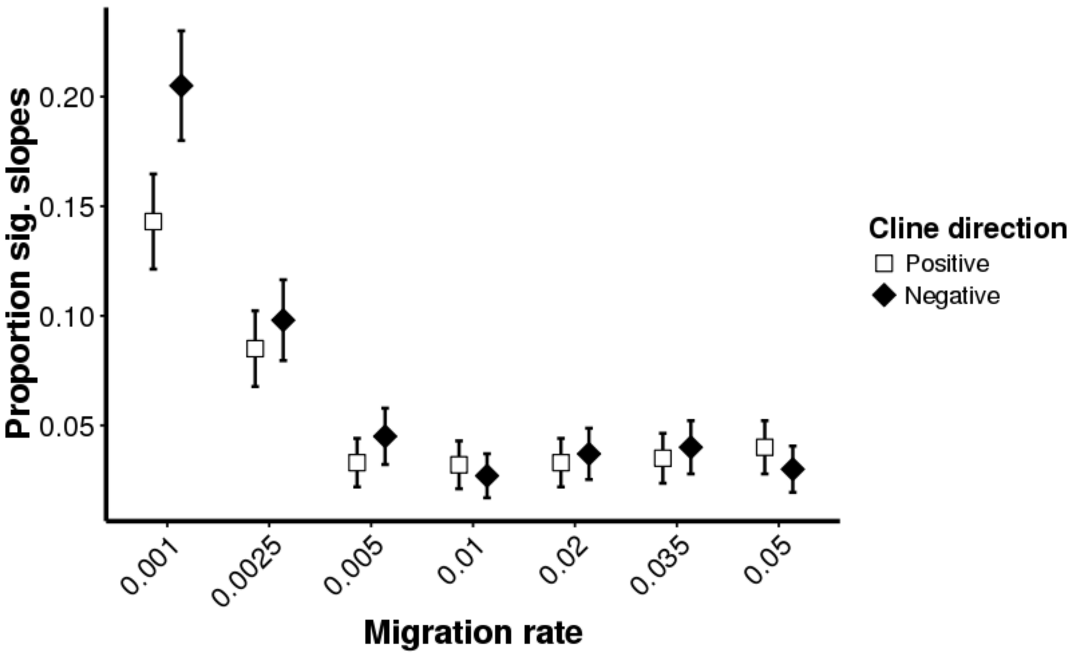
```
In [36]: #Subset to include only the last generation
         Bot_SlopeSum_Gen_999 <- subset(Bot_SlopeSum_Gen, Generation == 999)
```

```
#Plot mean slope by bottleneck proportion
plotMeanSlope_Bot_999 <- ggplot(Bot_SlopeSum_Gen_999, aes(x = factor(bot), y = mean)) +
    geom_errorbar(aes(ymin = mean - ci.lower, ymax = mean + ci.upper), width=0.15,
size=0.7,
    position = position_dodge(width = 0.5))+
    ylab("Mean slope") + xlab("Bottleneck proportion") + geom_point(size = 3.5, color =
"black")
plotMeanSlope_Bot_999 + ng1.45
```



```
In [37]: #Extrca columns that will be used to create melting dataset for plotting proportion of
         positive and negative slopes
         MigRate_SlopeSum_Gen_999_Melt <- dplyr::select(MigRate_SlopeSum_Gen_999, Mig_rate,
         prop_sigPos, ci.lower.Pos, ci.upper.Pos, prop_sigNeg,
           ci.lower.Neg, ci.upper.Neg)

         #Melt dataframe
         dm1 <- melt(MigRate_SlopeSum_Gen_999_Melt[,c("Mig_rate", "prop_sigPos", "ci.lower.Pos",
         "ci.upper.Pos")],
                     id=c("Mig_rate", "ci.lower.Pos", "ci.upper.Pos"))
         dm2 <- melt(MigRate_SlopeSum_Gen_999_Melt[,c("Mig_rate", "prop_sigNeg", "ci.lower.Neg",
         "ci.upper.Neg")],
                     id=c("Mig_rate", "ci.lower.Neg", "ci.upper.Neg"))

         #Rename columns
         setnames(dm1, old = c("ci.lower.Pos", "ci.upper.Pos"), new = c("ci.lower", "ci.upper"))
         setnames(dm2, old = c("ci.lower.Neg", "ci.upper.Neg"), new = c("ci.lower", "ci.upper"))

         #Merge melted dataframes
         MigRate_SlopeSum_Gen_999_Melt <- rbind(dm1, dm2)
```

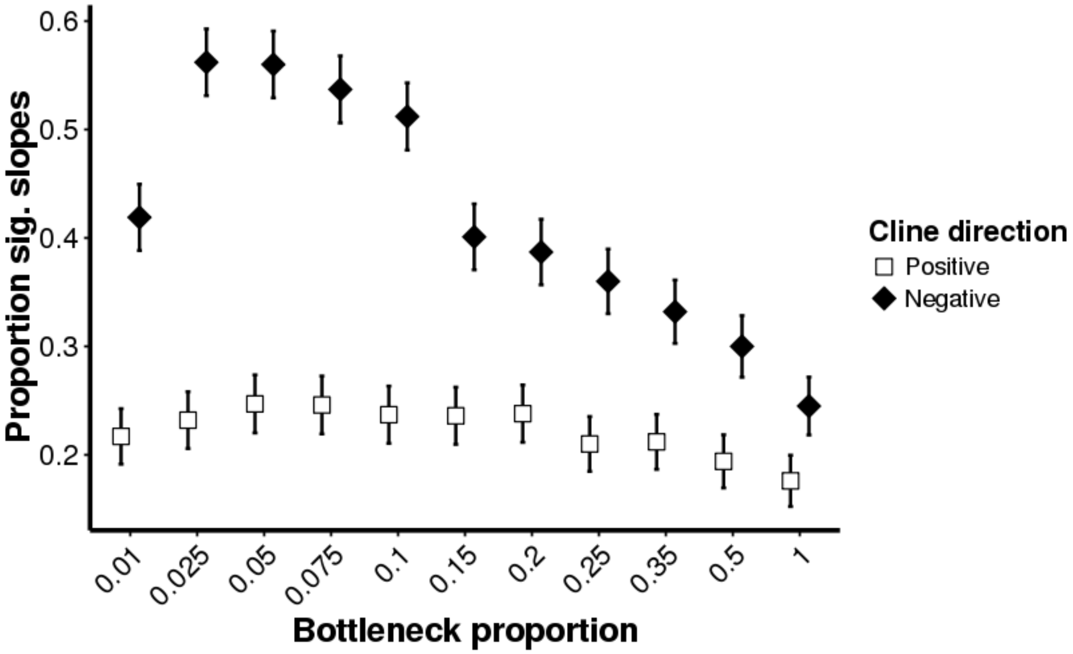
```
In [39]: #Plot proportion of significant slopes by migration rate
         plotPropSlope_Mig_rate_999 <- ggplot(MigRate_SlopeSum_Gen_999_Melt, aes(x =
         factor(Mig_rate), y = value,
             shape = variable,fill = variable)) +
             geom_errorbar(aes(ymin = value - ci.lower, ymax = value + ci.upper), width=0.15,
         size = 0.7,
             position = position_dodge(width = 0.55)) +
             ylab("Proportion sig. slopes") + xlab("Migration rate") + geom_point(size = 3.5,
         color = "black",
             position = position_dodge(width = 0.55)) +
             scale_shape_manual(labels = c("Positive", "Negative"),values=c(22, 23))+
             scale_fill_manual(labels = c("Positive", "Negative"),values=c("white", "black")) +
             labs(shape = 'Cline direction', fill = 'Cline direction')
         plotPropSlope_Mig_rate_999 + ng1.45
```



```
In [40]: #Extrca columns that will be used to create melting dataset for plotting proportion of
positive and negative slopes
         Bot_SlopeSum_Melt_Gen_999 <- dplyr::select(Bot_SlopeSum_Gen_999, bot, prop_sigPos,
ci.lower.Pos, ci.upper.Pos, prop_sigNeg,
           ci.lower.Neg, ci.upper.Neg)

         #Melt dataframe
         dm1 <- melt(Bot_SlopeSum_Melt_Gen_999[,c("bot", "prop_sigPos", "ci.lower.Pos",
"ci.upper.Pos")],
                     id=c("bot", "ci.lower.Pos", "ci.upper.Pos"))
         dm2 <- melt(Bot_SlopeSum_Melt_Gen_999[,c("bot", "prop_sigNeg", "ci.lower.Neg",
"ci.upper.Neg")],
                     id=c("bot", "ci.lower.Neg", "ci.upper.Neg"))

         #Rename columns
         setnames(dm1, old = c("ci.lower.Pos", "ci.upper.Pos"), new = c("ci.lower", "ci.upper"))
         setnames(dm2, old = c("ci.lower.Neg", "ci.upper.Neg"), new = c("ci.lower", "ci.upper"))
```

```
#Merge melted dataframes
Bot_SlopeSum_Melt_Gen_999 <- rbind(dm1, dm2)
```

```
In [41]: #Plot proportion of significant slopes by migration rate
         plotPropSlope_bot_999 <- ggplot(Bot_SlopeSum_Melt_Gen_999, aes(x = factor(bot), y =
         value, shape = variable,
             fill = variable)) +
             geom_errorbar(aes(ymin = value - ci.lower, ymax = value + ci.upper), width=0.15,
         size = 0.7,
             position = position_dodge(width = 0.55)) +
             ylab("Proportion sig. slopes") + xlab("Bottleneck proportion") + geom_point(size =
         3.5, color = "black",
             position = position_dodge(width = 0.55)) +
             scale_shape_manual(labels = c("Positive", "Negative"),values=c(22, 23))+
             scale_fill_manual(labels = c("Positive", "Negative"),values=c("white", "black")) +
             labs(shape = 'Cline direction', fill = 'Cline direction')
         plotPropSlope_bot_999 + ng1.45
```


Proportion sig. slopes
Bottleneck proportion
Cline direction
□ Positive
◆ Negative