Lobyerev V1

Itai van Rijn

February 24, 2019

## implementation of Lobyerev & Hoffman (2018) Selectivity model

This document describes the implementation of the selectivity algorithm described in Lobyrev, Feodor, and Matthew J. Hoffman. “A morphological and geometric method for estimating the selectivity of gill nets.” Reviews in Fish Biology and Fisheries 28.4 (2018): 909-924.

### Step 1: Import field data on fish catch and gill net properties

#### Step 1.1: Import catch data

Catch data structure: Column 1: ‘Mesh\_size’ - name of the net (later connected to net properties under the same name) Column 2: ‘Length\_group’ - Fish total length (cm) Column 3: ‘Wedged’ - Number of wedged individuals Column 4: ‘Tangled’ - Number of tangled individuals

Catch.data.Cod <- read.csv("~/kinneret modeling/selectivity/R code Feodor paper/Lobyerev-selectivity/Data/Catch data Cod.csv")  
#Show the first 6 lines:  
print(head(Catch.data.Cod))

## Mesh\_size Length\_group Wedged Tangled  
## 1 20 14 2 2  
## 2 20 16 20 5  
## 3 20 18 15 12  
## 4 20 20 11 8  
## 5 20 22 20 18  
## 6 20 24 32 24

#### Step 1.2: Import table of net properties

Net properties data structure: (See Figure 2 in paper) Column 1: ‘Mesh\_size’ - mesh size (knot to knot) (mm) Column 2: ‘y’ - Smaller of the angels between mesh threads

net.properties <- read.csv("~/kinneret modeling/selectivity/R code Feodor paper/Lobyerev-selectivity/Data/net properties.csv")  
#Show table:  
print(net.properties)

## Mesh\_size y  
## 1 20 60  
## 2 25 60  
## 3 30 60

#### Step 1.3: Add angel in Radian

net.properties$Radian=0.018\*net.properties$y  
#Show table:  
print(net.properties)

## Mesh\_size y Radian  
## 1 20 60 1.08  
## 2 25 60 1.08  
## 3 30 60 1.08

#### Step 1.4: Define the angel between the end of upper and lower jaws (phi)

phi\_deg=7  
phi\_radian=0.018\*phi\_deg

#### Step 1.5: Calculate the jaw length approximated by the linear function

slope\_jaw=1.02  
intecept\_jaw=3.41  
Catch.data.Cod$Jaw\_length=intecept\_jaw + (slope\_jaw\*Catch.data.Cod$Length\_group)  
#Calculate h  
Catch.data.Cod$h=Catch.data.Cod$Jaw\_length \* sin(phi\_radian) \* 2  
print(head(Catch.data.Cod))

## Mesh\_size Length\_group Wedged Tangled Jaw\_length h  
## 1 20 14 2 2 17.69 4.446094  
## 2 20 16 20 5 19.73 4.958815  
## 3 20 18 15 12 21.77 5.471535  
## 4 20 20 11 8 23.81 5.984256  
## 5 20 22 20 18 25.85 6.496977  
## 6 20 24 32 24 27.89 7.009698

#### Step 1.6: Merge the catch data with the net data

Catch.data.Cod=merge(Catch.data.Cod,net.properties,by="Mesh\_size")  
#print first 6 lines  
print(head(Catch.data.Cod))

## Mesh\_size Length\_group Wedged Tangled Jaw\_length h y Radian  
## 1 20 14 2 2 17.69 4.446094 60 1.08  
## 2 20 16 20 5 19.73 4.958815 60 1.08  
## 3 20 18 15 12 21.77 5.471535 60 1.08  
## 4 20 20 11 8 23.81 5.984256 60 1.08  
## 5 20 22 20 18 25.85 6.496977 60 1.08  
## 6 20 24 32 24 27.89 7.009698 60 1.08

### Step 2: Calculate P(O|C) and P(Th|C)

For calculating P(O|C) use eq. 4 in the paper

#Calculate sin,cos,tan  
Catch.data.Cod$sin=sin(0.5\*Catch.data.Cod$Radian)  
Catch.data.Cod$cos=cos(0.5\*Catch.data.Cod$Radian)  
Catch.data.Cod$tan=tan(0.5\*Catch.data.Cod$Radian)  
#Calculate P(O|C) Eq. 4  
Catch.data.Cod$POC=(((Catch.data.Cod$cos \* Catch.data.Cod$Mesh\_size)-Catch.data.Cod$h)\*((Catch.data.Cod$sin\*Catch.data.Cod$Mesh\_size)-(Catch.data.Cod$tan\*Catch.data.Cod$h)))/(Catch.data.Cod$cos\*Catch.data.Cod$sin\*(Catch.data.Cod$Mesh\_size)^2)  
#Calculate P(Th|c)  
Catch.data.Cod$PThC=1-Catch.data.Cod$POC  
#Print first 6 rows  
print(head(Catch.data.Cod))

## Mesh\_size Length\_group Wedged Tangled Jaw\_length h y Radian  
## 1 20 14 2 2 17.69 4.446094 60 1.08  
## 2 20 16 20 5 19.73 4.958815 60 1.08  
## 3 20 18 15 12 21.77 5.471535 60 1.08  
## 4 20 20 11 8 23.81 5.984256 60 1.08  
## 5 20 22 20 18 25.85 6.496977 60 1.08  
## 6 20 24 32 24 27.89 7.009698 60 1.08  
## sin cos tan POC PThC  
## 1 0.514136 0.8577087 0.5994296 0.5488078 0.4511922  
## 2 0.514136 0.8577087 0.5994296 0.5054167 0.4945833  
## 3 0.514136 0.8577087 0.5994296 0.4638123 0.5361877  
## 4 0.514136 0.8577087 0.5994296 0.4239946 0.5760054  
## 5 0.514136 0.8577087 0.5994296 0.3859636 0.6140364  
## 6 0.514136 0.8577087 0.5994296 0.3497193 0.6502807

### Step 3: Calculate P(W|E)

Import data on maximal girth (G\_max). In final version girth information will be part of the complete catch data

G\_max\_wide <- read.csv("~/kinneret modeling/selectivity/R code Feodor paper/Lobyerev-selectivity/Data/G\_max\_wide.csv", stringsAsFactors=FALSE)  
#Transform to long format  
G\_max\_long=gather(G\_max\_wide, t, G\_max, m1:m73, factor\_key=TRUE)  
#Arrange table  
#Drop NA  
G\_max\_long=G\_max\_long[!is.na(G\_max\_long$length\_group),]  
G\_max\_long=G\_max\_long[!is.na(G\_max\_long$G\_max),]  
#Drop the second column 't'  
G\_max\_long=G\_max\_long[ , !names(G\_max\_long) %in% c("t")]  
#Order by first column  
G\_max\_long=G\_max\_long[order(G\_max\_long$length\_group),]

For each net calculate P(W|E)

#Calculate the girth which each net will retain  
net.properties$girth\_retained=round(1.14\*4\*net.properties$Mesh\_size)  
#print(net.properties)  
#Calculate the P(W|E) for each length group. P(W|E) is the ratio between fish over net.properties$girth\_retained and the total number of fish in each group\_size for each net type  
PWE\_t=expand.grid(net.properties$Mesh\_size,unique(G\_max\_long$length\_group)) %>%   
 dplyr::rename(Mesh\_size = Var1,  
 length\_group = Var2) %>%  
 left\_join(net.properties) %>%   
 left\_join(G\_max\_long) %>%   
 select(Mesh\_size,girth\_retained,length\_group,G\_max) %>%   
 mutate(is.bigger = G\_max >= girth\_retained) %>%   
 group\_by(Mesh\_size,length\_group) %>%   
 summarize(PWE = sum(is.bigger)/n()) %>%  
 dplyr::rename(Length\_group=length\_group)

## Joining, by = "Mesh\_size"

## Joining, by = "length\_group"

PWE\_t=data.frame(PWE\_t)  
print(head(PWE\_t))

## Mesh\_size Length\_group PWE  
## 1 20 14 0.3333333  
## 2 20 16 0.7272727  
## 3 20 18 0.8333333  
## 4 20 20 0.8695652  
## 5 20 22 0.9814815  
## 6 20 24 1.0000000

Catch.data.Cod=merge(Catch.data.Cod,PWE\_t ,by=c("Mesh\_size", "Length\_group"))

### Step 4: Calculate Eq. 2

#### Step 4.1: Calculate P(E|O) by the linear equation

##Get the min and max size groups for each net  
#Create table  
net\_PEO=data.frame(Mesh\_size=unique(Catch.data.Cod$Mesh\_size))  
#Length group interval  
Length\_group\_interval=2  
#The minimal size for each net  
min\_wedged=data.frame(Catch.data.Cod %>%  
 filter(!is.na(Wedged)) %>%  
 group\_by(Mesh\_size) %>%  
 summarize(min\_size = min(Length\_group, na.rm = TRUE)))  
#The maximal size for each net  
max\_wedged=data.frame(Catch.data.Cod %>%  
 filter(!is.na(Wedged)) %>%  
 group\_by(Mesh\_size) %>%  
 summarize(max\_size = max(Length\_group, na.rm = TRUE)))  
#merge  
net\_PEO=merge(net\_PEO,min\_wedged,by="Mesh\_size")  
net\_PEO=merge(net\_PEO,max\_wedged,by="Mesh\_size")  
#Substract and add the Length group interval  
net\_PEO$min\_size=net\_PEO$min\_size-(Length\_group\_interval/2)  
net\_PEO$max\_size=net\_PEO$max\_size+(Length\_group\_interval/2)  
#Calculate the linear function  
net\_PEO$slope=-1/(net\_PEO$max\_size-net\_PEO$min\_size)  
net\_PEO$intercept=1-(net\_PEO$slope\*net\_PEO$min\_size)  
#print  
print(net\_PEO)

## Mesh\_size min\_size max\_size slope intercept  
## 1 20 13 31 -0.05555556 1.722222  
## 2 25 17 35 -0.05555556 1.944444  
## 3 30 21 37 -0.06250000 2.312500

#Calculate PEO  
#Merge slope and intercept to data  
Catch.data.Cod=merge(Catch.data.Cod,net\_PEO,by="Mesh\_size")  
Catch.data.Cod$PEO=Catch.data.Cod$intercept+(Catch.data.Cod$Length\_group)\*Catch.data.Cod$slope  
#Remove PEO values if no fish were wedged  
Catch.data.Cod[is.na(Catch.data.Cod$Wedged),"PEO"]=NA  
#Print first 6 lines  
print(head(Catch.data.Cod))

## Mesh\_size Length\_group Wedged Tangled Jaw\_length h y Radian  
## 1 20 14 2 2 17.69 4.446094 60 1.08  
## 2 20 16 20 5 19.73 4.958815 60 1.08  
## 3 20 18 15 12 21.77 5.471535 60 1.08  
## 4 20 20 11 8 23.81 5.984256 60 1.08  
## 5 20 22 20 18 25.85 6.496977 60 1.08  
## 6 20 24 32 24 27.89 7.009698 60 1.08  
## sin cos tan POC PThC PWE min\_size  
## 1 0.514136 0.8577087 0.5994296 0.5488078 0.4511922 0.3333333 13  
## 2 0.514136 0.8577087 0.5994296 0.5054167 0.4945833 0.7272727 13  
## 3 0.514136 0.8577087 0.5994296 0.4638123 0.5361877 0.8333333 13  
## 4 0.514136 0.8577087 0.5994296 0.4239946 0.5760054 0.8695652 13  
## 5 0.514136 0.8577087 0.5994296 0.3859636 0.6140364 0.9814815 13  
## 6 0.514136 0.8577087 0.5994296 0.3497193 0.6502807 1.0000000 13  
## max\_size slope intercept PEO  
## 1 31 -0.05555556 1.722222 0.9444444  
## 2 31 -0.05555556 1.722222 0.8333333  
## 3 31 -0.05555556 1.722222 0.7222222  
## 4 31 -0.05555556 1.722222 0.6111111  
## 5 31 -0.05555556 1.722222 0.5000000  
## 6 31 -0.05555556 1.722222 0.3888889

#### Step 4.2: Claculate Eq.2

Catch.data.Cod$Ntotal=Catch.data.Cod$Wedged/(Catch.data.Cod$POC\*Catch.data.Cod$PWE\*Catch.data.Cod$PEO)  
#Print first 6 lines  
print(head(Catch.data.Cod))

## Mesh\_size Length\_group Wedged Tangled Jaw\_length h y Radian  
## 1 20 14 2 2 17.69 4.446094 60 1.08  
## 2 20 16 20 5 19.73 4.958815 60 1.08  
## 3 20 18 15 12 21.77 5.471535 60 1.08  
## 4 20 20 11 8 23.81 5.984256 60 1.08  
## 5 20 22 20 18 25.85 6.496977 60 1.08  
## 6 20 24 32 24 27.89 7.009698 60 1.08  
## sin cos tan POC PThC PWE min\_size  
## 1 0.514136 0.8577087 0.5994296 0.5488078 0.4511922 0.3333333 13  
## 2 0.514136 0.8577087 0.5994296 0.5054167 0.4945833 0.7272727 13  
## 3 0.514136 0.8577087 0.5994296 0.4638123 0.5361877 0.8333333 13  
## 4 0.514136 0.8577087 0.5994296 0.4239946 0.5760054 0.8695652 13  
## 5 0.514136 0.8577087 0.5994296 0.3859636 0.6140364 0.9814815 13  
## 6 0.514136 0.8577087 0.5994296 0.3497193 0.6502807 1.0000000 13  
## max\_size slope intercept PEO Ntotal  
## 1 31 -0.05555556 1.722222 0.9444444 11.57589  
## 2 31 -0.05555556 1.722222 0.8333333 65.29266  
## 3 31 -0.05555556 1.722222 0.7222222 53.73527  
## 4 31 -0.05555556 1.722222 0.6111111 48.82138  
## 5 31 -0.05555556 1.722222 0.5000000 105.59213  
## 6 31 -0.05555556 1.722222 0.3888889 235.29074

### Step 5: Calculate CPUE

(table 21)

#Define the number of trials  
n\_trials=11  
#Aggregate catch for each mesh size  
Catch.data.Cod$CPUE=rowSums(Catch.data.Cod[,c("Wedged","Tangled")],na.rm=T)/n\_trials  
#Print first 6 lines  
print(head(Catch.data.Cod))

## Mesh\_size Length\_group Wedged Tangled Jaw\_length h y Radian  
## 1 20 14 2 2 17.69 4.446094 60 1.08  
## 2 20 16 20 5 19.73 4.958815 60 1.08  
## 3 20 18 15 12 21.77 5.471535 60 1.08  
## 4 20 20 11 8 23.81 5.984256 60 1.08  
## 5 20 22 20 18 25.85 6.496977 60 1.08  
## 6 20 24 32 24 27.89 7.009698 60 1.08  
## sin cos tan POC PThC PWE min\_size  
## 1 0.514136 0.8577087 0.5994296 0.5488078 0.4511922 0.3333333 13  
## 2 0.514136 0.8577087 0.5994296 0.5054167 0.4945833 0.7272727 13  
## 3 0.514136 0.8577087 0.5994296 0.4638123 0.5361877 0.8333333 13  
## 4 0.514136 0.8577087 0.5994296 0.4239946 0.5760054 0.8695652 13  
## 5 0.514136 0.8577087 0.5994296 0.3859636 0.6140364 0.9814815 13  
## 6 0.514136 0.8577087 0.5994296 0.3497193 0.6502807 1.0000000 13  
## max\_size slope intercept PEO Ntotal CPUE  
## 1 31 -0.05555556 1.722222 0.9444444 11.57589 0.3636364  
## 2 31 -0.05555556 1.722222 0.8333333 65.29266 2.2727273  
## 3 31 -0.05555556 1.722222 0.7222222 53.73527 2.4545455  
## 4 31 -0.05555556 1.722222 0.6111111 48.82138 1.7272727  
## 5 31 -0.05555556 1.722222 0.5000000 105.59213 3.4545455  
## 6 31 -0.05555556 1.722222 0.3888889 235.29074 5.0909091

### Step 6: Calculate Nw per hour

(table 23)

#Define number of hours of single (??) field trial  
n\_hours=12  
Catch.data.Cod$Ntotal\_per\_hour=Catch.data.Cod$Ntotal/n\_hours  
#Print first 6 lines  
print(head(Catch.data.Cod))

## Mesh\_size Length\_group Wedged Tangled Jaw\_length h y Radian  
## 1 20 14 2 2 17.69 4.446094 60 1.08  
## 2 20 16 20 5 19.73 4.958815 60 1.08  
## 3 20 18 15 12 21.77 5.471535 60 1.08  
## 4 20 20 11 8 23.81 5.984256 60 1.08  
## 5 20 22 20 18 25.85 6.496977 60 1.08  
## 6 20 24 32 24 27.89 7.009698 60 1.08  
## sin cos tan POC PThC PWE min\_size  
## 1 0.514136 0.8577087 0.5994296 0.5488078 0.4511922 0.3333333 13  
## 2 0.514136 0.8577087 0.5994296 0.5054167 0.4945833 0.7272727 13  
## 3 0.514136 0.8577087 0.5994296 0.4638123 0.5361877 0.8333333 13  
## 4 0.514136 0.8577087 0.5994296 0.4239946 0.5760054 0.8695652 13  
## 5 0.514136 0.8577087 0.5994296 0.3859636 0.6140364 0.9814815 13  
## 6 0.514136 0.8577087 0.5994296 0.3497193 0.6502807 1.0000000 13  
## max\_size slope intercept PEO Ntotal CPUE  
## 1 31 -0.05555556 1.722222 0.9444444 11.57589 0.3636364  
## 2 31 -0.05555556 1.722222 0.8333333 65.29266 2.2727273  
## 3 31 -0.05555556 1.722222 0.7222222 53.73527 2.4545455  
## 4 31 -0.05555556 1.722222 0.6111111 48.82138 1.7272727  
## 5 31 -0.05555556 1.722222 0.5000000 105.59213 3.4545455  
## 6 31 -0.05555556 1.722222 0.3888889 235.29074 5.0909091  
## Ntotal\_per\_hour  
## 1 0.9646579  
## 2 5.4410550  
## 3 4.4779389  
## 4 4.0684481  
## 5 8.7993439  
## 6 19.6075616

### Step 7:

#### Step 7.1: SLl,t for each length class as Ntotal-Qt

(table 24)

Catch.data.Cod$SL\_l\_t=Catch.data.Cod$Ntotal\_per\_hour-Catch.data.Cod$CPUE  
#Print first 6 lines  
print(head(Catch.data.Cod))

## Mesh\_size Length\_group Wedged Tangled Jaw\_length h y Radian  
## 1 20 14 2 2 17.69 4.446094 60 1.08  
## 2 20 16 20 5 19.73 4.958815 60 1.08  
## 3 20 18 15 12 21.77 5.471535 60 1.08  
## 4 20 20 11 8 23.81 5.984256 60 1.08  
## 5 20 22 20 18 25.85 6.496977 60 1.08  
## 6 20 24 32 24 27.89 7.009698 60 1.08  
## sin cos tan POC PThC PWE min\_size  
## 1 0.514136 0.8577087 0.5994296 0.5488078 0.4511922 0.3333333 13  
## 2 0.514136 0.8577087 0.5994296 0.5054167 0.4945833 0.7272727 13  
## 3 0.514136 0.8577087 0.5994296 0.4638123 0.5361877 0.8333333 13  
## 4 0.514136 0.8577087 0.5994296 0.4239946 0.5760054 0.8695652 13  
## 5 0.514136 0.8577087 0.5994296 0.3859636 0.6140364 0.9814815 13  
## 6 0.514136 0.8577087 0.5994296 0.3497193 0.6502807 1.0000000 13  
## max\_size slope intercept PEO Ntotal CPUE  
## 1 31 -0.05555556 1.722222 0.9444444 11.57589 0.3636364  
## 2 31 -0.05555556 1.722222 0.8333333 65.29266 2.2727273  
## 3 31 -0.05555556 1.722222 0.7222222 53.73527 2.4545455  
## 4 31 -0.05555556 1.722222 0.6111111 48.82138 1.7272727  
## 5 31 -0.05555556 1.722222 0.5000000 105.59213 3.4545455  
## 6 31 -0.05555556 1.722222 0.3888889 235.29074 5.0909091  
## Ntotal\_per\_hour SL\_l\_t  
## 1 0.9646579 0.6010215  
## 2 5.4410550 3.1683278  
## 3 4.4779389 2.0233934  
## 4 4.0684481 2.3411754  
## 5 8.7993439 5.3447984  
## 6 19.6075616 14.5166525

#### Step 7.2: N\_AP calculation

First the table of Nlim (table 22c) is defined, in reality it is in input

!!!Notice- value of tau is very sensitive to the sum of the CPUE

###The next data frame will be an experimental input  
n\_lim=data.frame(Mesh\_size=c(20,25,30),Nlim=c(36,32,12))  
#Sum CPUE per net  
CPUE=data.frame(Catch.data.Cod %>%  
 group\_by(Mesh\_size) %>%  
 summarize(CPUE\_sum = sum(CPUE, na.rm = TRUE)))  
#merge to n\_lim table  
n\_lim=merge(n\_lim,CPUE,by="Mesh\_size")  
#Calculate tau (table 25)  
n\_lim$tau=n\_hours/(-log(1-(n\_lim$CPUE/n\_lim$Nlim)))  
n\_lim$N\_AP=((n\_lim$Nlim\*(1.71))/n\_lim$tau)\*n\_hours  
print(n\_lim)

## Mesh\_size Nlim CPUE\_sum tau N\_AP  
## 1 20 36 25.09091 10.050904 73.49787  
## 2 25 32 26.27273 6.974731 94.14557  
## 3 30 12 11.72727 3.171088 77.65157

#### Step 7.3: Size specific N\_AP

!!!In table 26 you use data from table 24 (SL)- is SL=0 if there is no value for this length group in table 24?

#Replace NA's in column 'SL\_l\_t' with 0  
Catch.data.Cod[is.na(Catch.data.Cod$SL\_l\_t),"SL\_l\_t"]=0  
#merge   
Catch.data.Cod=merge(Catch.data.Cod,n\_lim,by="Mesh\_size")  
#Calculate N\_AP per size class  
Catch.data.Cod$N\_AP\_size=(Catch.data.Cod$N\_AP\*(Catch.data.Cod$CPUE/Catch.data.Cod$CPUE\_sum))+Catch.data.Cod$SL\_l\_t  
print(head(Catch.data.Cod))

## Mesh\_size Length\_group Wedged Tangled Jaw\_length h y Radian  
## 1 20 14 2 2 17.69 4.446094 60 1.08  
## 2 20 16 20 5 19.73 4.958815 60 1.08  
## 3 20 18 15 12 21.77 5.471535 60 1.08  
## 4 20 20 11 8 23.81 5.984256 60 1.08  
## 5 20 22 20 18 25.85 6.496977 60 1.08  
## 6 20 24 32 24 27.89 7.009698 60 1.08  
## sin cos tan POC PThC PWE min\_size  
## 1 0.514136 0.8577087 0.5994296 0.5488078 0.4511922 0.3333333 13  
## 2 0.514136 0.8577087 0.5994296 0.5054167 0.4945833 0.7272727 13  
## 3 0.514136 0.8577087 0.5994296 0.4638123 0.5361877 0.8333333 13  
## 4 0.514136 0.8577087 0.5994296 0.4239946 0.5760054 0.8695652 13  
## 5 0.514136 0.8577087 0.5994296 0.3859636 0.6140364 0.9814815 13  
## 6 0.514136 0.8577087 0.5994296 0.3497193 0.6502807 1.0000000 13  
## max\_size slope intercept PEO Ntotal CPUE  
## 1 31 -0.05555556 1.722222 0.9444444 11.57589 0.3636364  
## 2 31 -0.05555556 1.722222 0.8333333 65.29266 2.2727273  
## 3 31 -0.05555556 1.722222 0.7222222 53.73527 2.4545455  
## 4 31 -0.05555556 1.722222 0.6111111 48.82138 1.7272727  
## 5 31 -0.05555556 1.722222 0.5000000 105.59213 3.4545455  
## 6 31 -0.05555556 1.722222 0.3888889 235.29074 5.0909091  
## Ntotal\_per\_hour SL\_l\_t Nlim CPUE\_sum tau N\_AP N\_AP\_size  
## 1 0.9646579 0.6010215 36 25.09091 10.0509 73.49787 1.666208  
## 2 5.4410550 3.1683278 36 25.09091 10.0509 73.49787 9.825743  
## 3 4.4779389 2.0233934 36 25.09091 10.0509 73.49787 9.213402  
## 4 4.0684481 2.3411754 36 25.09091 10.0509 73.49787 7.400811  
## 5 8.7993439 5.3447984 36 25.09091 10.0509 73.49787 15.464070  
## 6 19.6075616 14.5166525 36 25.09091 10.0509 73.49787 29.429263

### Step 8: Calculate selectivity

(table 28)

##Calculate selectivity  
Catch.data.Cod$selectivity=Catch.data.Cod$CPUE/Catch.data.Cod$Ntotal\_per\_hour  
##plot  
Catch.data.Cod$Mesh\_size\_fac=as.character(Catch.data.Cod$Mesh\_size)  
ggplot(Catch.data.Cod, aes(x=Length\_group, y=selectivity, group=Mesh\_size\_fac)) +  
 geom\_line(aes(color=Mesh\_size\_fac))+  
 geom\_point(aes(color=Mesh\_size\_fac))+  
 labs(x = "length Group",y="Selectivity",color="Mesh Size")

## Warning: Removed 19 rows containing missing values (geom\_path).

## Warning: Removed 19 rows containing missing values (geom\_point).

