

reuniao__DaniloMori__29mar

Mori, Danilo Pereira

27 de março de 2018

Olá caros,

Segue um resumo do que tenho feito. Na primeira parte do arquivo apresento o texto que resume a metodologia de conversão de parâmetros que utilizei. Então sigo apresentando os resultados dessas conversões de parâmetros para o modelo EI e EE com graficos exploratórios apenas para mostrar a relação entre as variáveis, sem fazer nenhuma análise a priori. Na terceira parte eu realizo uma análise preliminar das variáveis do modelo EI: GOF, m, theta, U e kernel_percentil (%k). Segue um resumo da história

-Na conversão dos parâmetros eu fiquei em dúvida na correspondência entre “d” (a mean dispersal) da aproximação de Chisholm and Linchestein 2009 (eqn 2) e o “Sigma” da distribuição de LaPlace, que estamos usando na função de dispersão. Essa dúvida ganhou força pois a maioria dos ‘m’{EE}, aqueles que obtive a partir dos parâmetros de EE, apresentam valores por vezes muito superiores ao teto de valor teórico da variável que é 1 (Hubbell 2001).

-Da conversão de m{EI}, parece que trabalhos cuja amostra são muito grandes (ou seja J muito grande) distoam dos demais valores estimados de kernel{EI} e consequentemente de kernel_percentil{EI}. Aqui vale lembrar que kernel é o valor utilizado como sigma na função de LaPlace e kernel_percentil é a correspondente chuva de propágulos. Isso foi observado no caminho inverso de kernel{EE} para m{EI} também.

-Da análise preliminar das variáveis do modelo EI, incluindo os parâmetros estimados originais e os convertidos a partir destes, resumo 5 fatos: a) GOF{EI} não demonstra efeito pronunciável da cobertura vegetal, sendo o modelo nulo mais plausível com média ~95 b) o mesmo vale para m{EI}, sendo a média (intercepto) deste ~ 0.697 c) theta{EI} demonstra efeito da cobertura vegetal sendo este efeito positivo d) U{EI} apresenta efeito negativo da cobertura vegetal, junto com theta{EI} a seleção dessas duas variáveis foram as únicas onde o modelo nulo não ficou dentro do intervalo de plausibilidade e) kernel_percentil{EI} não demonstra efeito pronunciável da cobertura vegetal, sendo o modelo nulo mais plausível com média ~ 99%

-Considerando o fato (e) e que a única informação informada a priori no modelo EE foi kernel de dispersão um possível caminho não seria utilizar para comparar os dois modelos? Explicando: poderíamos utilizar as simulações EE que consideram o cenário de limitação à dispersão mais severo para comparar com o modelo EI, que também apresentou alta severidade na limitação à dispersão? Isso claro, se a conversão de valores estiver certa.

Conversão de parâmetros entre modelo neutro de espaço implícito (EI) e de espaço explícito (EE)

Os dois modelos neutros EE e EI são implementações da teoria geral desenvolvida por Hubbell 2001, contudo, a relação entre seus parâmetros não é direta. Aqui convertemos os parâmetros obtidos por verossimilhança utilizando o modelo EI nos parâmetros do modelo EE, U (taxa de especiação na paisagem) e d (média da dispersão). E convertemos o parâmetro U do modelo EE, estimado utilizando uma formula semi-analítica baseada na natureza genealogica do modelo coalescente dado a riqueza observada e a função de dispersão (parâmetro informado a priori), em theta; e a partir da função de dispersão e forma obtemos m. Para fazer a conversão entre U e theta utilizamos a definição presente em Hubbell 2001 onde:

$$theta := 2 * JM * U;$$

Onde JM = número de indivíduos na paisagem = área total do recorte de paisagem (500 ha) X densidade observada (DA) X cobertura vegetal observada (p). Utilizamos essa relação para conversão reciproca de theta e U:

$$\begin{aligned} \theta &= 2 * 500 * DA * p * U \\ U &= \theta / (2 * 500 * DA * p) \end{aligned}$$

Para a conversão de m e d utilizamos a aproximação de caso geral apresentado por Chisholm & Linchestein (2009). Essa aproximação considera que existe apenas duas escalas: a amostra de tamanho J e uma paisagem infinita e homogênea. Em sua aproximação geral a forma da função de dispersão (ou seja, a distribuição de probabilidade que descreve a probabilidade de colonização de um propágulo em função da distância da planta progenitora) não têm influência no resultado da aproximação. Contudo, a área da amostra, relativo à média de dispersão, e a forma da amostra têm consideráveis influências na aproximação. Amostras circulares apresentam maior acurácia do que amostra retangulares e quanto maior o diâmetro da amostra em relação à média de dispersão maior a acurácia da aproximação. Em nosso caso, selecionamos apenas trabalhos fitossociológicos que realizaram amostras em bloco único e em nossas simulações aproximamos essas amostras como um bloco único quadrado. Assim, a aproximação geral apresentada por Chisholm & Linchestein 2009:

$$m \sim P d / \pi A;$$

onde P = perímetro da amostra, d = mean dispersal distance, A = área da amostra

pode ser reescrita como:

$$m \sim 4l d / \pi l^2;$$

Onde l = lado da amostra = \sqrt{A} e $A = J/DA$ (número de indivíduos da amostra dividido pela densidade observada na amostra)

Simplificando:

$$m \sim 4 d / \pi \sqrt{J/DA}$$

A fim de estabelecer uma relação recíproca entre os parâmetros utilizamos a aproximação enquanto igualdade e temos:

$$m = 4 * d / (\pi * \sqrt{J/DA})$$

$$d = m * \pi * \sqrt{J/DA} / 4$$

A função de dispersão que utilizamos no modelo EE tem como distribuição de probabilidade subjacente a distribuição de LaPlace. Essa distribuição possui dois parâmetros, média e escalar. A distribuição de LaPlace pode ser entendida como uma distribuição exponencial negativa espelhada, assim, sua média não é equivalente à distância média de dispersão, essa informação está associada com o parâmetro escalar da distribuição.

Uma vez calculado d a partir do modelo EI, estimamos a equivalente chuva de propágulos que definimos em termos de porcentagem de propágulos que se mantêm até uma determinada distância da planta progenitora. Aqui vale relembrar uma nota sobre a função que criamos para realizar tal tarefa:

“Há um erro que é proporcionalmente maior quando as distâncias são pequenas. Acho que é arredondamento que está na função de quantil, que faz ela andar aos saltos. Mas a boa notícia é que o erro é o mesmo para todos os kernels. Então, apesar da função não achar exatamente o sigma necessário para que p indivíduos caiam até distance de distância, o sigma encontrado resulta na mesma distância (quantil), o que torna os sigmas calculados com os mesmos parâmetros p e distance equivalentes.”

A distância de referência é $\text{dist}_0 = 100/\sqrt{DA}$ (lado da célula da simulação). Assim, tanto o perímetro/área da amostra quando a distância média da função de dispersão estão na mesma escala métrica. Portanto a conversão pode ser feita diretamente.

Resultado da conversão dos parâmetros

A conversão dos parâmetros de EI para os parâmetros de EE ocorre em duas etapas: primeiro estimamos U e d utilizando as formulas supracitadas, então a partir de d estimamos a correspondente chuva de propágulos (%k). %k é a porcentagem de propágulos que se manteve até a distância de referência (dist_0) considerando o parâmetro de dispersão ‘d’ estimado pela aproximação de Chisholm & Linchestein (2009).

O arquivo que estou mantendo toda a criação e auditoria dos dados é 'criacao_e_auditoria_dados.Rmd', o resultado dos dados dos dois modelos e da conversão dos modelos está na tabela 'df_resultados.txt'. Para vocês poderem reproduzir o código chave vou partir das colunas chave dessa tabela.

Conversão de parâmetros modelo EI

```
# SdT: i) filtrar colunas chave de df_resultados ii) converter theta -> U e
# m -> d; iii) calcular a correspondente chuva de propágulos

# i) filtrar colunas chave # colunas: SiteCode, p, S, DA, J, GOF, theta, m
df_EI <- df_resultados %>% filter(modelo == "EI") %>% .[, c(1, 3:6, 9:11)]

# ii) conversão dos parâmetros # conversão dos parâmetros
df_EI %<>% mutate(U = theta/(2 * 500 * DA * p), kernel = m * pi * (sqrt(J/DA))/4)

# iii) Calculando a correspondente chuva de propágulos # A função a seguir é
# uma modificação de qkernel de utility_functions, que o PI desenvolveu.
# Essa função tem como inputs: sigma e a distância de referência e como
# output o percentil determinado pelo número de pontos que se acumulam até a
# determinada distância dado o sigma.

# funcao para calcular a porcentagem da chuva de propagulos a partir de
# kernel e
f_percentil.kernel <- function(i, df_ = df_EI) {
  percentilkernel <- function(sigma, density, npoints = 1e+05) {
    # metro na escala da simulacao
    d_ind_MA <- 100/sqrt(density) #tanto a metrica da paisagem quanto a distancia de referencia
    b_laplace <- sigma/sqrt(2) ## relação entre sigma e b, a variância da laplaciana
    ## gerando valores de uma distribuição laplaciana e convertendo para a
    ## distância em metros
    X_laplace <- d_ind_MA * round(rlaplace(npoints, s = b_laplace)/d_ind_MA)
    Y_laplace <- d_ind_MA * round(rlaplace(npoints, s = b_laplace)/d_ind_MA) # idem para Y
    dist_laplace <- sqrt(X_laplace^2 + Y_laplace^2) #distâncias dos pontos até a origem
    percentil <- length(dist_laplace[dist_laplace <= d_ind_MA])/length(dist_laplace) # percentil
    return(percentil)
  }
  kernel_percentil <- percentilkernel(sigma = df_[i, "kernel"], density = df_[i,
    "DA"])
  return(kernel_percentil)
}
df_EI$kernel_percentil <- sapply(1:nrow(df_EI), f_percentil.kernel) #armazenando
```

-> Pontos a confirmar: i) a maneira que utilizei 'd', da aproximação de Chisholm & Linchestein 2009, está correta dentro do uso de sigma?

Danilo: eu havia pensando que 'd' era a variância (ou escalar, correto?) da distribuição de LaPlace, mas não sei se entendi corretamente a correspondência entre os parâmetros estimados pela aproximação(Chisholm & Linchestein, 2009) e pela distribuição de LaPlace.

ii) A função para calcular o percentil está correta?

-> Gráfico Exploratório

```
car::scatterplotMatrix(~theta + U + m + kernel + kernel_percentil, reg.line = "",
  smoother = "", data = df_EI, main = "")
```

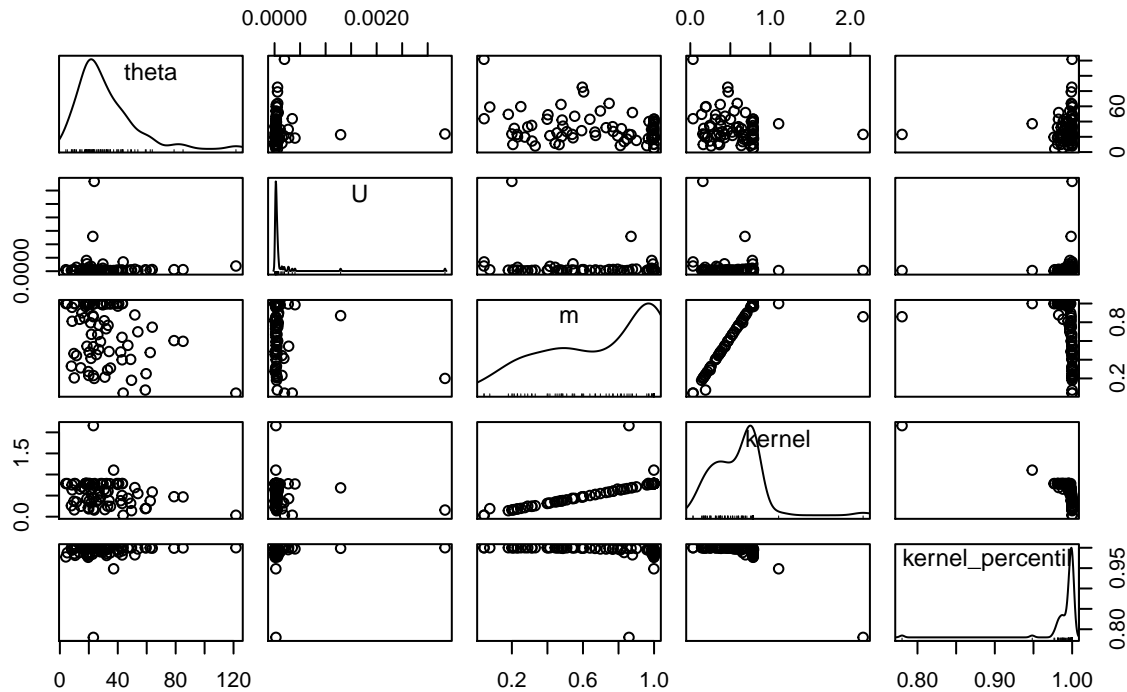


figura 1. Relação entre os parametros originais (theta e m) e as estimativas a partir deles (U, kernel e kernel_percentil).

A partir da relação entre theta e U parece existir dois pontos que distoam da nuvem de dados geral; olhando a relação entre m e kernel/kernel percentil também nota-se dois pontos distoando do padrão geral dos dados. E pela relação geral das variáveis parece que esses pontos não são os mesmos (gráfico U ~ kernel_percentil). Vamos ver quem são esses Site:

```
df_EI %>% filter(U > 0.001 | kernel_percentil < 0.96)
```

```
##   SiteCode      p      S      DA      J GOF   theta      m
## 1 MGuberl3 0.018041724 85 975.000 975 100 22.75907 0.8706217
## 2 MGuberl5 0.007424379 68 957.000 957 100 23.72298 0.2000630
## 3 SPeec1 0.743179321 144 1182.129 12105 80 23.13184 0.8590295
## 4 SPpei1 0.969241860 165 1545.455 3060 100 37.13857 0.9988957
##           U      kernel kernel_percentil
## 1 1.293814e-03 0.6837847           0.99861
## 2 3.338853e-03 0.1571291           1.00000
## 3 2.633005e-05 2.1589767           0.78003
## 4 2.479344e-05 1.1039327           0.94837
```

As variáveis que podem influenciar as conversões são p, DA e J, além das variáveis de interesse m e theta. Em um primeiro palpite diria que os U distoantes ocorrem por estarem em paisagens muito fragmentadas; já no caso de kernel, deve ser por conta do valor de J. Para comparação segue o summary dessas variáveis

```
df_EI[, 1:5] %>% summary
```

```
##   SiteCode      p      S      DA
## Length:80      Min.    :0.007424      Min.    : 26.00      Min.    : 595
```

```
## Class :character 1st Qu.:0.291633 1st Qu.: 73.75 1st Qu.:1138
## Mode :character Median :0.672751 Median : 98.50 Median :1474
## Mean :0.602527 Mean :105.85 Mean :1497
## 3rd Qu.:0.921663 3rd Qu.:134.00 3rd Qu.:1745
## Max. :1.000000 Max. :230.00 Max. :3148
## J
## Min. : 595
## 1st Qu.: 1163
## Median : 1488
## Mean : 1792
## 3rd Qu.: 1803
## Max. :12105
```

De fato os p dos maiores Us estão nas paisagens mais fragmentadas, já o m decorre dos altos valores de J. o SiteCode que apresenta J ~ 12 mil distoa consideravelmente da maioria dos demais trabalhos, como podemos observar pela média, media e 3o quartil de J. Me parece que talvez fosse um ponto a se avaliar a relevância de se mante-lo.

Conversão de parâmetros modelo EE

```
# SdT: i) info de interesse; ii) conversão de parâmetros

# i) info de interesse #
df_EE <- df_resultados %>% filter(modelo == "EE") %>% .[, 1:9]
# ii) conversão de parâmetros
df_EE %<>% mutate(theta = 2 * 500 * DA * p * U, m = 4 * kernel/pi * sqrt(J/DA)) # estou considerando q
```

-> Pontos a confirmar: i) a questão da relação de 'd' da aproximação e do parâmetro da LaPlace é válido aqui

Segue summary dos dados:

```
df_EE %>% summary

## SiteCode kernel_percentil p S
## Length:960 Min. :0.2500 Min. :0.007424 Min. : 26.00
## Class :character 1st Qu.:0.5875 1st Qu.:0.291633 1st Qu.: 73.75
## Mode :character Median :0.7250 Median :0.672751 Median : 98.50
## Mean :0.7075 Mean :0.602527 Mean :105.85
## 3rd Qu.:0.8625 3rd Qu.:0.921663 3rd Qu.:134.00
## Max. :0.9900 Max. :1.000000 Max. :230.00
## DA J kernel U
## Min. : 595 Min. : 595 Min. :0.4592 Min. :0.0007575
## 1st Qu.:1138 1st Qu.: 1163 1st Qu.:1.2599 1st Qu.:0.0049397
## Median :1474 Median : 1488 Median :1.7983 Median :0.0080076
## Mean :1497 Mean : 1792 Mean :2.0074 Mean :0.0090700
## 3rd Qu.:1745 3rd Qu.: 1803 3rd Qu.:2.4203 3rd Qu.:0.0122275
## Max. :3148 Max. :12105 Max. :7.4934 Max. :0.0319706
## GOF theta m
## Min. : 0.00 Min. : 55.46 Min. : 0.5846
## 1st Qu.: 88.00 1st Qu.: 1976.86 1st Qu.: 1.6160
## Median : 98.00 Median : 6220.27 Median : 2.3373
## Mean : 87.28 Mean : 8805.60 Mean : 2.7302
## 3rd Qu.:100.00 3rd Qu.:13560.88 3rd Qu.: 3.1592
## Max. :100.00 Max. :46932.83 Max. :22.3550
```

Daqui vejo um grande problema da conversão dos parâmetros no modelo EE, m por definição deve variar

em $[0;1]$ e a grande maioria dos valores está acima desse intervalo. Isso me faz questionar se eu fiz a correspondência correta entre ‘d’ da aproximação e o parâmetro escala da LaPlace (como supracitado), contudo, isso me motivou também a ajustar o modelo de Etienne às SADs{EE} para ter uma base de comparação. Eu ainda não analisei o theta e m que obtive desses ajustes mas a tabela resultante está no git “df_SAD_EE-parEI-viaEI.txt”.

```
car::scatterplotMatrix(~U + theta + kernel_percentil + m, reg.line = "", smoother = "",
  data = df_EE, main = "Parâmetros modelo EE")
```

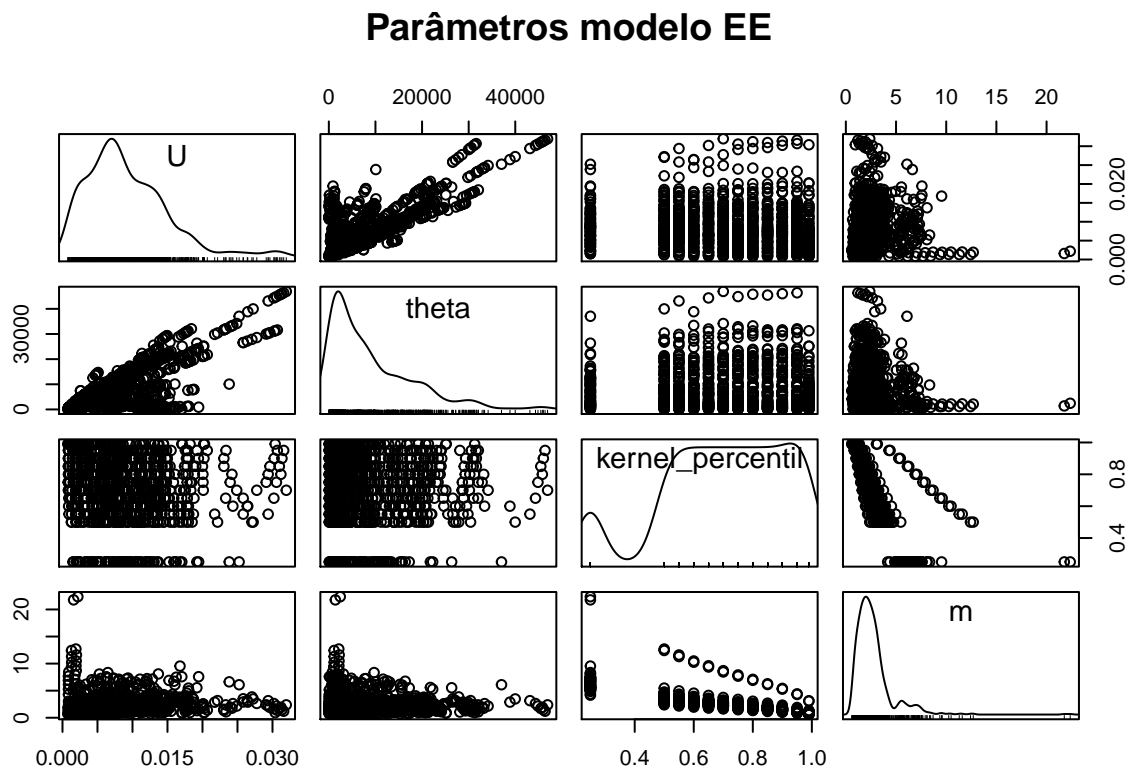


figura 2. Relação entre os parametros originais (theta e m) e as estimativas a partir deles (U, kernel e kernel_percentil).

Em uma primeira avaliação me parece que existe um problema com ‘m’, em todas as classes de kernel_percentil tem valores que distoam do padrão geral dos pontos sendo sistematicamente superiores, a diferença entre o padrão da nuvem e esses pontos diminui com o aumento de kernel_percentil. Olhando os gráficos de (theta,U) ~ m parece que há valores distoando em theta e U também e eles parecem estar relacionados com ‘m’. Pelo o que observamos na sessão anterior, especulo que esses pontos devem estar associados com o tamanho da amostra (J). Vou começar avaliando qual os pontos com maior ‘m’

```
df_EE %>% mutate(k = factor(kernel_percentil)) %>% ggplot(aes(x = k, y = m)) +
  geom_boxplot() + geom_text(aes(label = SiteCode))
```

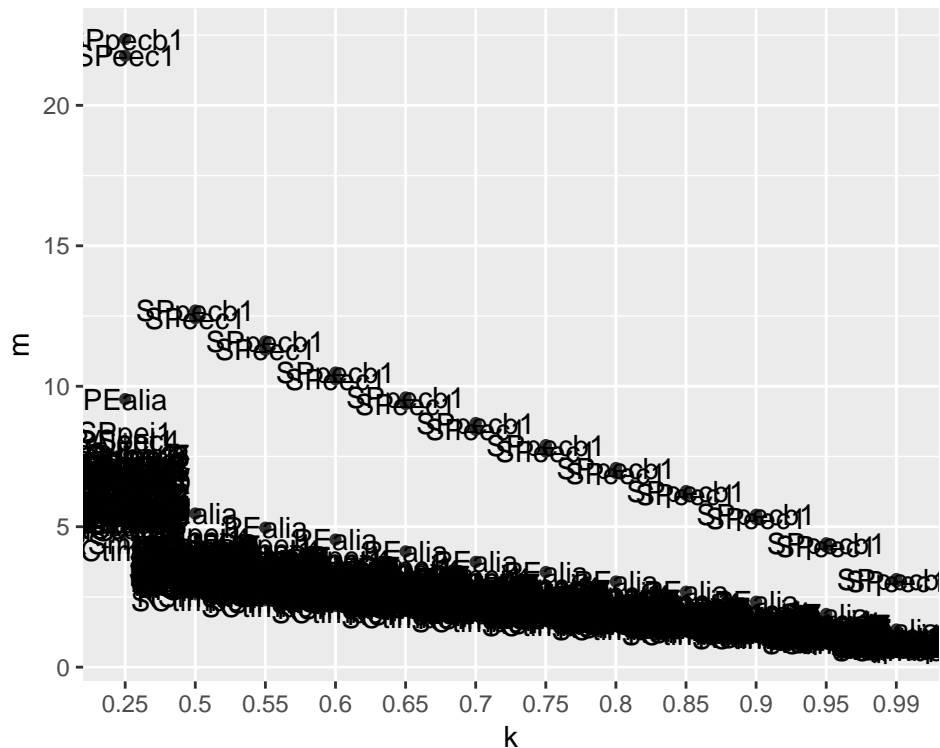


figura 3. Relação entre 'm' e função de dispersão, os pontos estão representados pelos códigos utilizados para identificar as diferentes observações (SiteCode)

Os pontos que distoam em 'm' são sistematicamente os mesmos, vou avaliar quais são as características desses SiteCodes:

```
df_EE %>% filter(SiteCode %in% c("SPee1", "SPp1")) %>% .[, c(1, 3:6)] %>%
  unique
```

```
##   SiteCode      p    S      DA    J
## 1   SPee1 0.7431793 144 1182.129 12105
## 13  SPp1 0.9922531 221 1123.730 11507
```

Esses dois SiteCodes apresentam valores altos de J, o quê explica o aumento sistematico de 'm' em relação aos demais SiteCodes. Nesse aspecto, o mesmo problema que tivemos com a conversão de EI para EE, obtivemos no caminho inverso: J grandes tem efeito no parâmetro calculado.

Análise preliminar das variáveis do modelo EI

A seguir resumo os achados das análises preliminares que fiz com os resultados do modelo EI. Eu achei que seria mais fácil eu começar compartimentalizando a análise, não vou mostrar todos os passos que fiz apenas a seleção dos modelos mais adequados e seus gráficos diagnósticos.

GOF

-> análise exploratória

```
plot(GOF ~ p, data = df_EI)
```

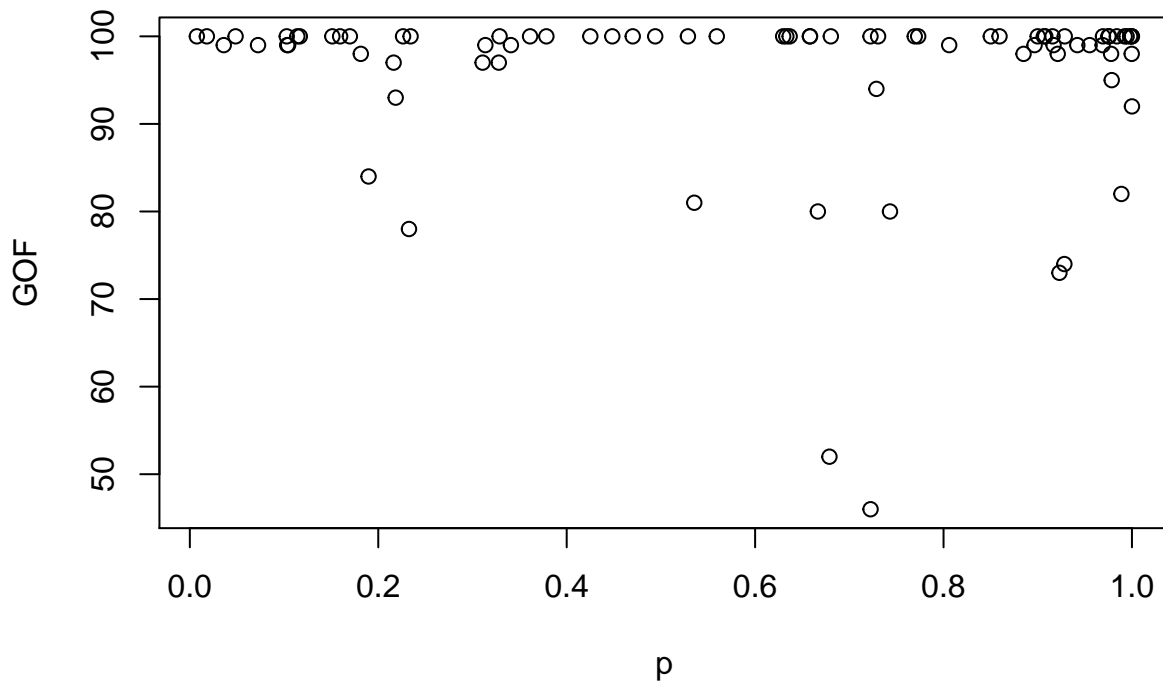


figura 4. Relação entre GOF ('Goodness-of-fit') e p (% cobertura vegetal)

a seguir seleção dos modelos que podem descrever a possível relação entre as variáveis. Em um primeiro momento utilizei a distribuição binomial mas obtive melhores resultados utilizando a função normal:

```
# l_md <- vector('list', length = 10) names(l_md) <-
# c('bin_logit', 'bin_logit_null', 'normal', 'normal_null',
# 'Gamma_inv', 'Gamma_inv_null', 'Gamma_id', 'Gamma_id_null', 'Gamma_log', 'Gamma_log_null')
# l_md[[1]] <- glm(cbind(GOF, 100-GOF) ~ p, family = 'binomial', df_EI)
# l_md[[2]] <- glm(cbind(GOF, 100-GOF) ~ 1, family = 'binomial', df_EI)
# l_md[[3]] <- glm(GOF ~ p, family = 'gaussian', df_EI) l_md[[4]] <- glm(GOF ~
# 1, family = 'gaussian', df_EI) l_md[[5]] <- glm(GOF ~ p, family =
# 'Gamma' (link=inverse), df_EI) l_md[[6]] <- glm(GOF ~ 1, family =
# 'Gamma' (link=inverse), df_EI) l_md[[7]] <- glm(GOF ~ p, family =
# 'Gamma' (link=identity), df_EI) l_md[[8]] <- glm(GOF ~ 1, family =
# 'Gamma' (link=identity), df_EI) l_md[[9]] <- glm(GOF ~ p, family =
# 'Gamma' (link=log), df_EI) l_md[[10]] <- glm(GOF ~ 1, family =
# 'Gamma' (link=log), df_EI) AICctab(l_md, weights = TRUE)
```

Os modelos mais plausíveis são aqueles que consideram a distribuição normal e o modelo que considera a influência de p em GOF está dentro do intervalo de plausibilidade mais soma menos da metade do peso de evidência do modelo mais plausível o modelo nulo. Segue os coeficientes estimados dos modelos mais plausíveis:

```
# sapply(l_md[4:3], coef)
```

O intercepto ou a média dos valores de GOF é similar para os dois modelos (~ 95). Curiosamente o modelo que considera o efeito da cobertura vegetal apresenta relação contrária à expectativa teórica: efeito negativo; esperava-se que com o aumento da cobertura houvesse também o aumento de GOF. Contudo esses valores

podem estar sendo influenciados pelos valores distoantes de $GOF < 70$. Com isso podemos estabelecer o

Fato 1: $GOF\{EI\}$ apresenta altos valores sempre e o possível efeito do gradiente de cobertura vegetal seria negativo.

m - probabilidade de imigração

‘m’ é definido como a probabilidade de um indivíduo da metacomunidade migrar para a comunidade local por evento de morte e como tal pode variar apenas entre 0 e 1. Primeiro, segue uma análise exploratória da variável:

```
plot(m ~ p, data = df_EI)
```

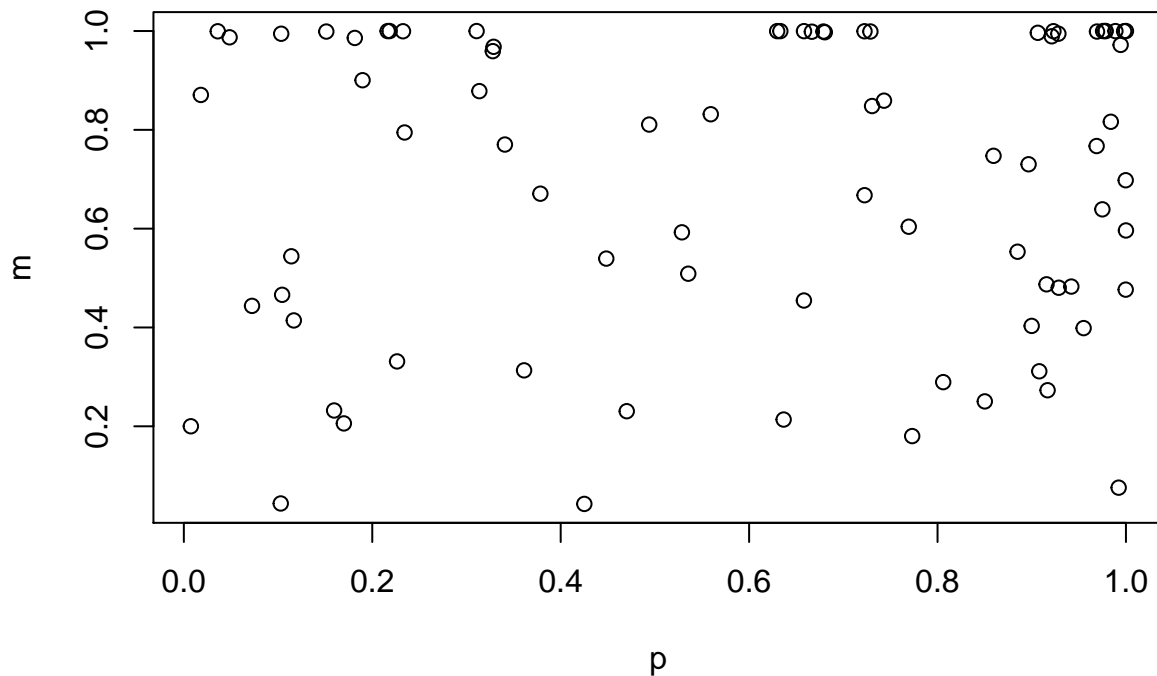


figura 5. Relação de $m \sim p$

A variável apresenta grande variância ao longo de seus possíveis valores, apesar de parecer existir um acúmulo de pontos ao redor de ~ 1 , não é possível afirmar a priori que existe algum efeito da fragmentação. Vou utilizar as distribuições normal e Gamma como distribuições subjacentes para construir os modelos.

```
# l_md <- vector('list', length = 8) names(l_md) <-
# c('normal', 'normal_null',
# 'Gamma_inv', 'Gamma_inv_null', 'Gamma_id', 'Gamma_id_null', 'Gamma_log', 'Gamma_log_null')
# l_md[[1]] <- glm(m ~ p, family = 'gaussian', df_EI) l_md[[2]] <- glm(m ~
# 1, family = 'gaussian', df_EI) l_md[[3]] <- glm(m ~ p, family =
# 'Gamma'(link=inverse), df_EI) l_md[[4]] <- glm(m ~ 1, family =
# 'Gamma'(link=inverse), df_EI) l_md[[5]] <- glm(m ~ p, family =
# 'Gamma'(link=identity), df_EI) l_md[[6]] <- glm(m ~ 1, family =
# 'Gamma'(link=identity), df_EI) l_md[[7]] <- glm(m ~ p, family =
```

```
# 'Gamma'(link=log),df_EI) l_md[[8]] <- glm(m ~ 1,family =
# 'Gamma'(link=log),df_EI) AICctab(l_md, weights = TRUE)
```

Os dois modelos dentro do intervalo de plausibilidade são aqueles que utilizam a distribuição normal como subjacente. O modelo mais plausível acumulando 0.71 peso de evidência foi o modelo nulo, o modelo que considera o efeito da fragmentação soma apenas 0.29 de peso de evidência e está bem próximo do limite de plausibilidade. Primeiro vou olhar para a estimativa dos parâmetros desses dois modelos:

```
# sapply(l_md[2:1],coef)
```

A estimativa do intercepto é bem próxima entre os dois modelos; o efeito da cobertura vegetal no segundo modelo mais plausível é bem sutil e positiva o que vem de encontro com a expectativa de quanto menor a fragmentação menor o isolamento da comunidade local (ou da amostra que estamos simulando).

Fato 2: a fragmentação não apresenta efeito pronunciável em ‘m’, o modelo que considera o efeito da cobertura vegetal, que é o segundo mais plausível, apresenta inclinação ~ 0.058 . A partir do modelo nulo, modelo mais plausível, a média de valores (ou intercepto) estimado é ~ 0.697 . Gostaria de avaliar a dispersão dos valores ao redor dos valores para comparação com o R quadrado dos modelos usados para análise dos resultados dos modelos EE [METAS GERAIS].

Theta - número fundamental da biodiversidade

Theta é um parâmetro adimensional que é definido como a composição do tamanho da metacomunidade, no qual a comunidade local é amostrada, e a taxa de especiação. Segue análise exploratória

```
plot(theta ~ p, data = df_EI)
```

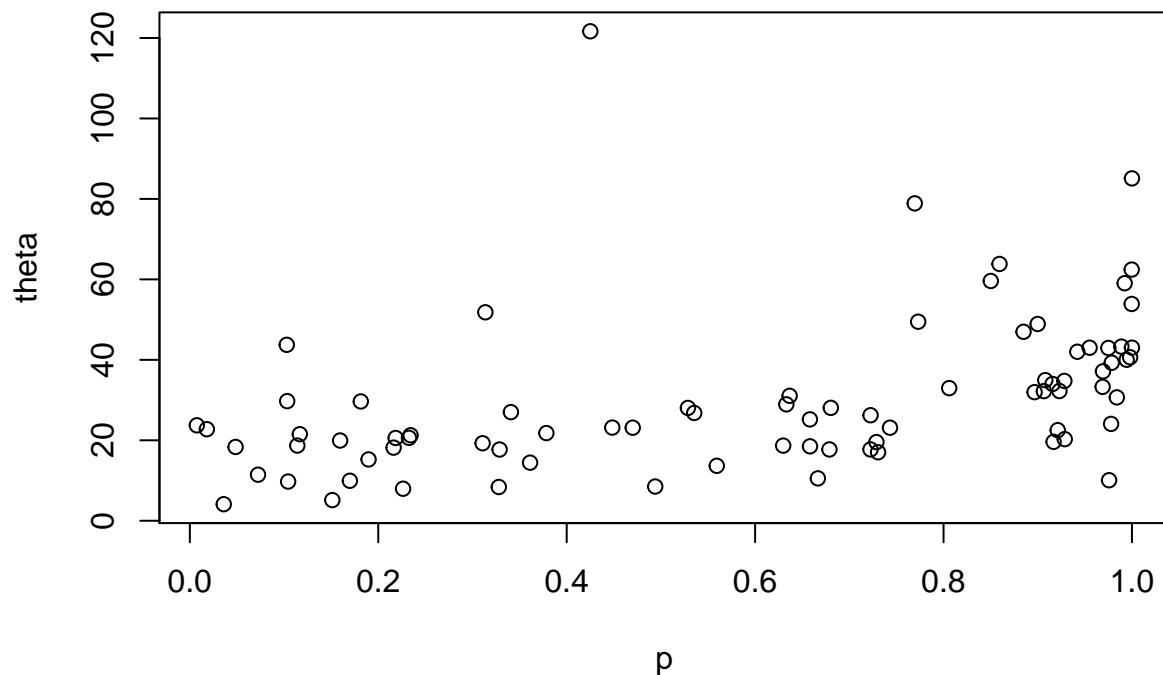


figura 6. $\theta \sim p$

A variável parece apresentar uma certa tendência positiva com a cobertura vegetal. Para modelar esta variável vou utilizar as mesmas distribuições subjacentes da variável 'm'.

```
# l_md <- vector('list', length = 8) names(l_md) <-
# c('normal', 'normal_null',
# 'Gamma_inv', 'Gamma_inv_null', 'Gamma_id', 'Gamma_id_null', 'Gamma_log', 'Gamma_log_null')
# l_md[[1]] <- glm(theta ~ p, family = 'gaussian', df_EI) l_md[[2]] <-
# glm(theta ~ 1, family = 'gaussian', df_EI) l_md[[3]] <- glm(theta ~ p, family
# = 'Gamma'(link=inverse), df_EI) l_md[[4]] <- glm(theta ~ 1, family =
# 'Gamma'(link=inverse), df_EI) l_md[[5]] <- glm(theta ~ p, family =
# 'Gamma'(link=identity), df_EI) l_md[[6]] <- glm(theta ~ 1, family =
# 'Gamma'(link=identity), df_EI) l_md[[7]] <- glm(theta ~ p, family =
# 'Gamma'(link=log), df_EI) l_md[[8]] <- glm(theta ~ 1, family =
# 'Gamma'(link=log), df_EI) AICctab(l_md, weights = TRUE)
```

Os modelos com distribuição Gamma e que consideram o efeito da cobertura vegetal na variável são os únicos dentro do intervalo de plausibilidade. O peso de evidência mostra que a qualidade de ajuste dos três é bem próxima, sendo aquele que utiliza a função de ligação inversa o mais plausível. Segue o sumário do modelo mais plausível:

```
# summary(l_md[[3]])
```

Como a função de ligação é inversa a leitura das estimativas não é na mesma escala da plotada na análise exploratória da variável. Podemos avaliar que existe um efeito positivo da cobertura vegetal em theta, para avaliar a estimativa do intercepto é necessário transformar para a escala da função de ligação. Isso vai de encontro com as expectativas: quanto maior a fragmentação => menor a metacomunidade; menor o tamanho das populações => maior as taxas de extinção - mesmo considerando que a fragmentação não altere o padrão biogeográfico da taxa de especiação certamente podemos considerar que vai diminuir o tamanho da metacomunidade e portanto theta, uma vez, que $\theta := 2JmU$ Fato 3: A cobertura vegetal apresenta efeito positivo em theta.

Variáveis Convertidas: U e %k

i) U - taxa de especiação

```
plot(U ~ p, data = df_EI)
```

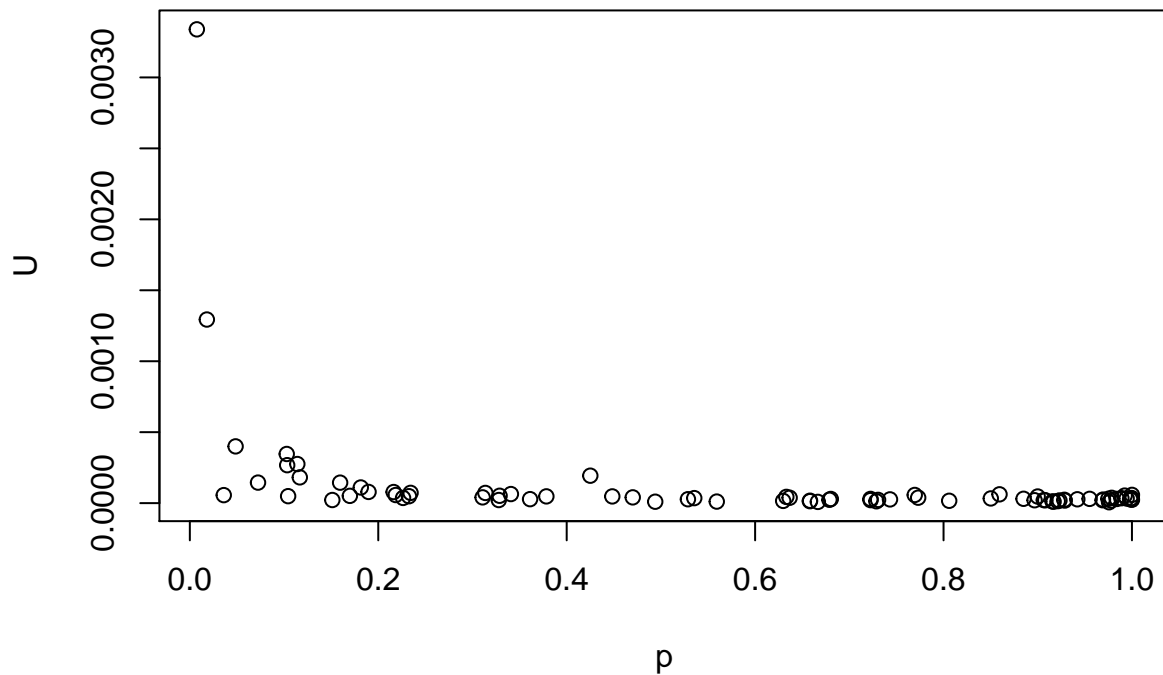


figura 8. $U\{EI\} \sim p$

Parece que há uma tendência de diminuição de U com o aumento de p , essa diminuição acompanha diminuição da variância.

```
# l_md <- vector('list', length = 8) names(l_md) <-
# c('normal', 'normal_null',
# 'Gamma_inv', 'Gamma_inv_null', 'Gamma_id', 'Gamma_id_null', 'Gamma_log', 'Gamma_log_null')
# l_md[[1]] <- glm(U ~ p, family = 'gaussian', df_EI) l_md[[2]] <- glm(U ~
# 1, family = 'gaussian', df_EI) l_md[[3]] <- glm(U ~ p, family =
# 'Gamma' (link=inverse), df_EI) l_md[[4]] <- glm(U ~ 1, family =
# 'Gamma' (link=inverse), df_EI) l_md[[5]] <- glm(U ~ p, family =
# 'Gamma' (link=identity), df_EI) l_md[[6]] <- glm(U ~ 1, family =
# 'Gamma' (link=identity), df_EI) l_md[[7]] <- glm(U ~ p, family =
# 'Gamma' (link=log), df_EI) l_md[[8]] <- glm(U ~ 1, family =
# 'Gamma' (link=log), df_EI) AICctab(l_md, weights = TRUE)
```

O modelo gama com função de ligação inversa onde considera a influência da cobertura vegetal foi o único modelo dentro do intervalo de plausibilidade e acumula todo o peso de evidência dos dados. Segue

Em código os gráficos diagnósticos, eu diria que o modelo mais plausível está realizando um ajuste bom. Segue o sumário do modelo:

```
# summary(l_md[[3]])
```

O modelo descreve uma relação negativa entre as variáveis, que vem de encontro com a análise exploratória. Novamente lembrando que a escala do modelo e do gráfico exploratório não coincidem. Futuramente irei trabalhar melhor em cada modelo.

Fato 4: $U\{EI\}$ diminui com o aumento da cobertura vegetal

ii) %k - kernel percentil

```
plot(kernel_percentil ~ p, data = df_EI)
```

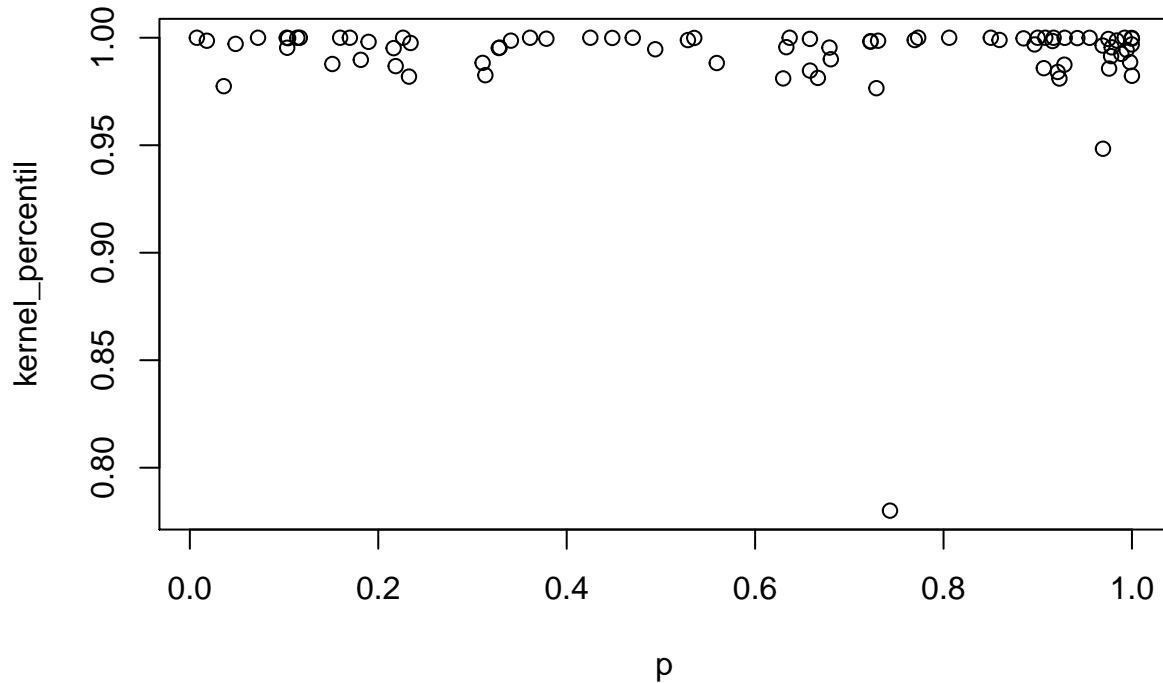


Figura 9. kernel percentil ~ p

Diria que não há uma tendência dos valores de %k variarem ao longo de p, me parece que de maneira geral a conversão de 'm' para %k apresenta cenários de alta severidade na limitação à dispersão dos indivíduos.

```
# l_md <- vector('list', length = 8) names(l_md) <-
# c('normal','normal_null',
# 'Gamma_inv','Gamma_inv_null','Gamma_id','Gamma_id_null','Gamma_log','Gamma_log_null')
# l_md[[1]] <- glm(kernel_percentil ~ p,family = 'gaussian',df_EI) l_md[[2]]
# <- glm(kernel_percentil ~ 1,family = 'gaussian',df_EI) l_md[[3]] <-
# glm(kernel_percentil ~ p,family = 'Gamma'(link=inverse),df_EI) l_md[[4]]
# <- glm(kernel_percentil ~ 1,family = 'Gamma'(link=inverse),df_EI)
# l_md[[5]] <- glm(kernel_percentil ~ p,family =
# 'Gamma'(link=identity),df_EI) l_md[[6]] <- glm(kernel_percentil ~ 1,family
# = 'Gamma'(link=identity),df_EI) l_md[[7]] <- glm(kernel_percentil ~
# p,family = 'Gamma'(link=log),df_EI) l_md[[8]] <- glm(kernel_percentil ~
# 1,family = 'Gamma'(link=log),df_EI) AICctab(l_md, weights = TRUE)
```

O modelo mais plausível é aquele que desconsidera o efeito da cobertura vegetal em kernel percentil, ainda dentro do intervalo de plausibilidade está o modelo que considera tal efeito. Vamos ver quais os coeficientes destes modelos:

```
# sapply(l_md[2:1],coef)
```

A estimativa do intercepto dos dois modelos é similar ~ 0.99%, o modelo que considera o efeito de p mostra

um efeito sutil e negativo. Sem levar em conta o possível valor distoante discutido na sessão da conversão de dados, me parece razoável afirmar que o efeito de p pode ser desprezível para efeitos práticos.

fato 5: %k{EI} apresenta valor médio ~ 0.99% ao longo de todo o gradiente de cobertura vegetal, que por sua vez tem efeito sutil ou mesmo desprezível na variável.

Reunião 29 de Abril

Email PI: Oi Danilo

Bem legais os resultados. Fiquei super curioso com a comparação entre os valores de theta, U e m obtidos pelos dois modelos. Vc pode fazer scatterplots para cada um destes parâmetros (cada ponto seria um plot) pra amanhã? Ahrs

-> Ideia de gráfico: $VR \sim \%k * \text{modelo} + (\sim p_class)$

descrição: Para cada uma das variáveis de interesse (VR; theta, U e m) vou produzir um scatterplot cuja variável independente será %k, os gráficos serão separados por classes de cobertura vegetal e os pontos serão coloridos pelo SiteCode e a forma deles pelo modelo (EI ou EE)

-> alternativa: $VR \sim p * \text{modelo} + (\sim \%k)$

descrição: para cada $VR \sim p$ e dividiria por classe de %k, o problema nesse caso é que as observações do modelo EI iriam aparecer apenas em um dos gráficos. A primeira opção permite melhor comparação.

```
df_resultados <- read.table(file = "/home/danilo/Documentos/Doutorado/artigo_mestrado/Rmd_e_dados/df_res
  header = T, as.is = T)
## Preparação dos dados ##
df_resultados %<>% mutate(p_class = cut(p, 12), k = 1 - kernel_percentil)
df_resultados %>% head
```

```
##   SiteCode kernel_percentil      p      S      DA      J      kernel      U GOF
## 1   BAjuss           0.25 0.859421 182 1198 1198 5.293956 0.01592218 100
## 2   BAjuss           0.50 0.859421 182 1198 1198 3.021432 0.02705866 100
## 3   BAjuss           0.55 0.859421 182 1198 1198 2.751210 0.02582330 100
## 4   BAjuss           0.60 0.859421 182 1198 1198 2.496593 0.02639610 100
## 5   BAjuss           0.65 0.859421 182 1198 1198 2.285264 0.02581922 100
## 6   BAjuss           0.70 0.859421 182 1198 1198 2.070920 0.02775694  99
##      theta      m modelo      p_class      k
## 1 16393.26 6.740474    EE (0.835,0.917] 0.75
## 2 27859.23 3.847006    EE (0.835,0.917] 0.50
## 3 26587.32 3.502950    EE (0.835,0.917] 0.45
## 4 27177.07 3.178761    EE (0.835,0.917] 0.40
## 5 26583.12 2.909689    EE (0.835,0.917] 0.35
## 6 28578.17 2.636778    EE (0.835,0.917] 0.30
```

Theta

```
df_resultados %>% ggplot(aes(x = k, y = log(theta), colour = SiteCode, shape = modelo)) +
  geom_point(aes(shape = modelo)) + geom_line() + facet_wrap(~p_class, ncol = 3,
  scales = "free") + theme(legend.position = "none")
```

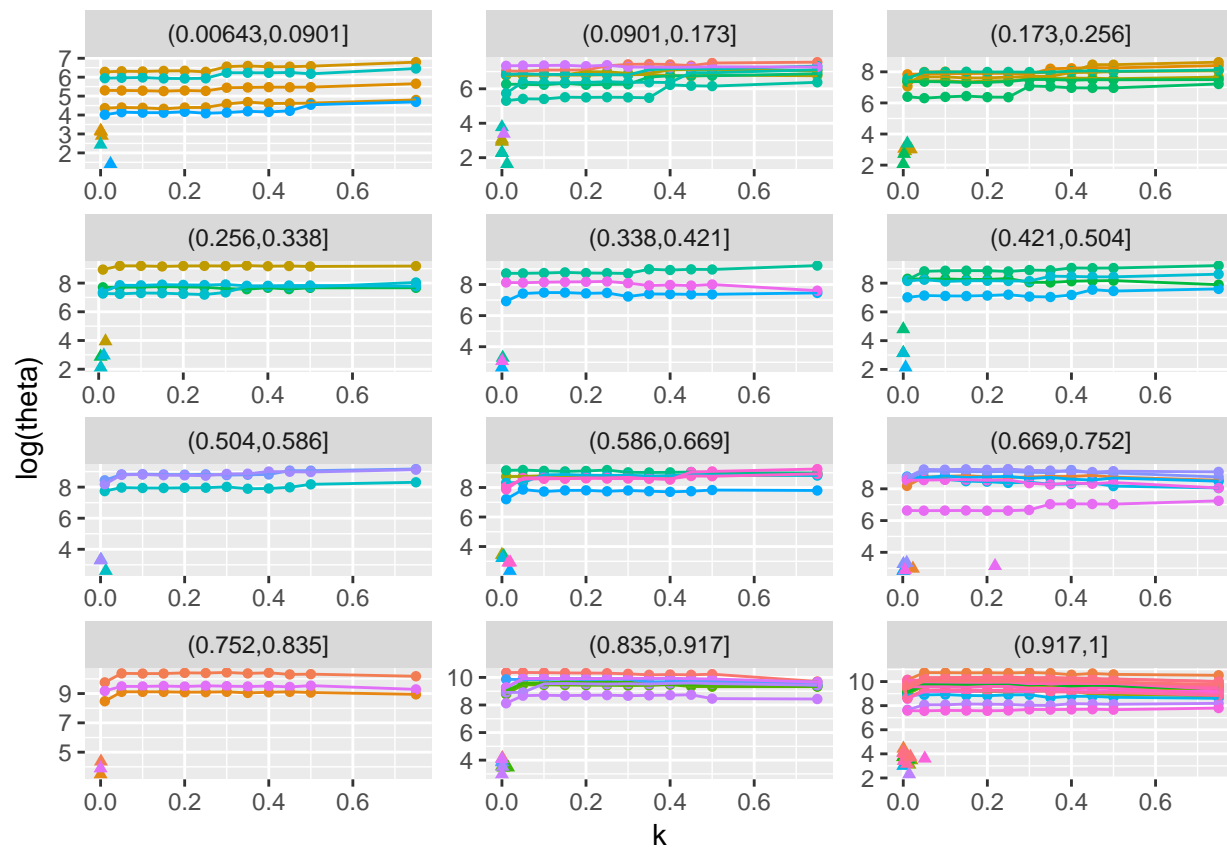


figura 10. $\log(\theta) \sim \%k$ ($\sim p_class$)

```
df_resultados %>% ggplot(aes(x = k, y = theta, colour = SiteCode, shape = modelo)) +
  geom_point(aes(shape = modelo)) + geom_line() + facet_wrap(~p_class, ncol = 3,
  scales = "free") + theme(legend.position = "none")
```

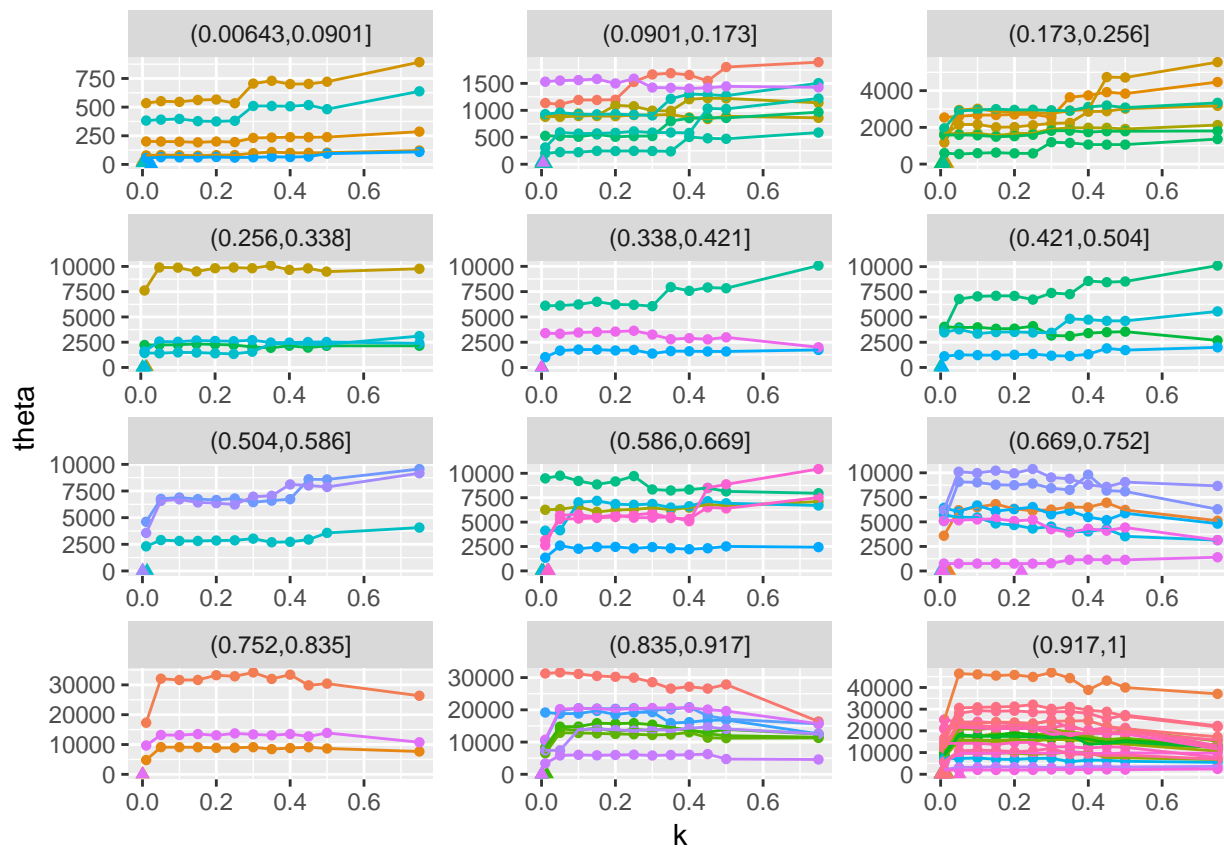


figura 11. $\theta \sim \%k$ ($\sim p_{\text{class}}$)

U

```
df_resultados %>% ggplot(aes(x = k, y = log(U), colour = SiteCode, shape = modelo)) +
  geom_point(aes(shape = modelo)) + geom_line() + facet_wrap(~p_class, ncol = 3,
  scales = "free") + theme(legend.position = "none")
```

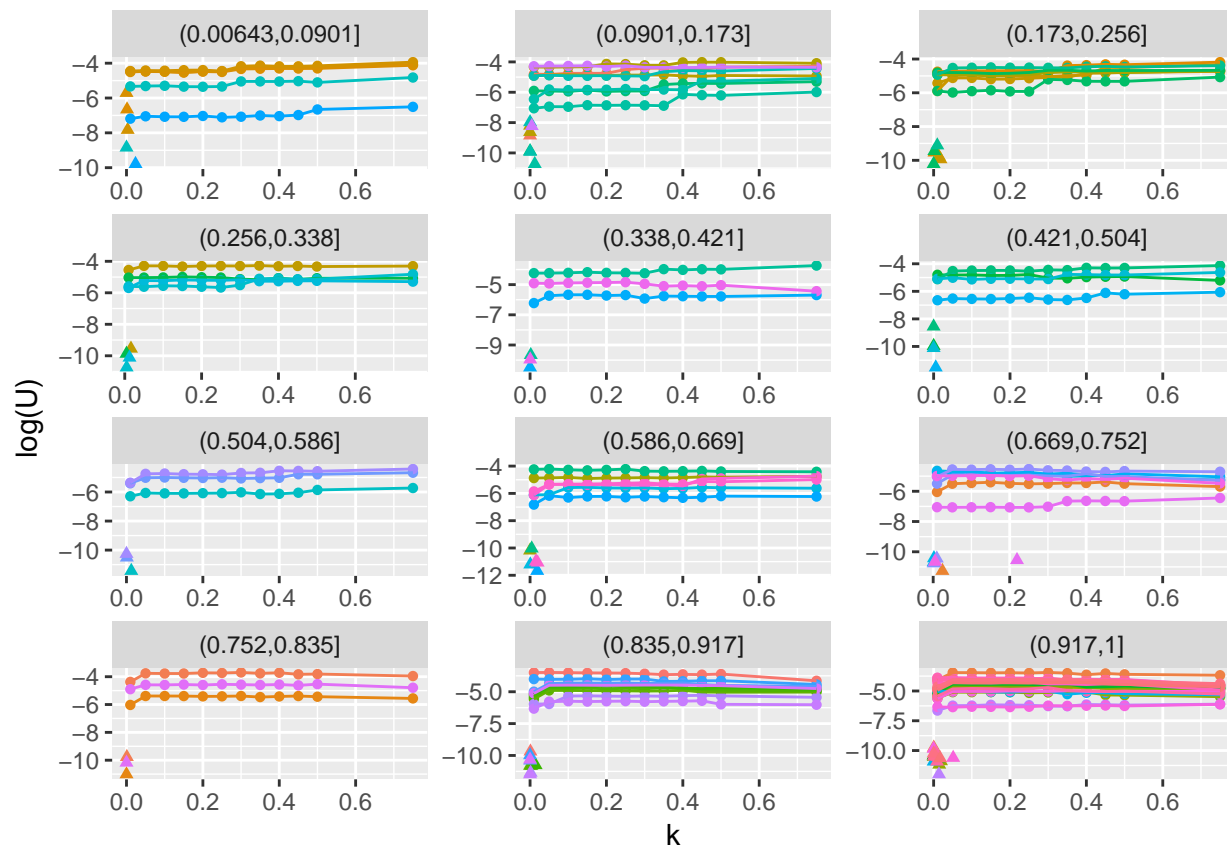



figura 12. $\log(U) \sim \%k$ ($\sim p_{\text{class}}$)

```
df_resultados %>% ggplot(aes(x = k, y = U, colour = SiteCode, shape = modelo)) +
  geom_point(aes(shape = modelo)) + geom_line() + facet_wrap(~p_class, ncol = 3,
    scales = "free") + theme(legend.position = "none")
```

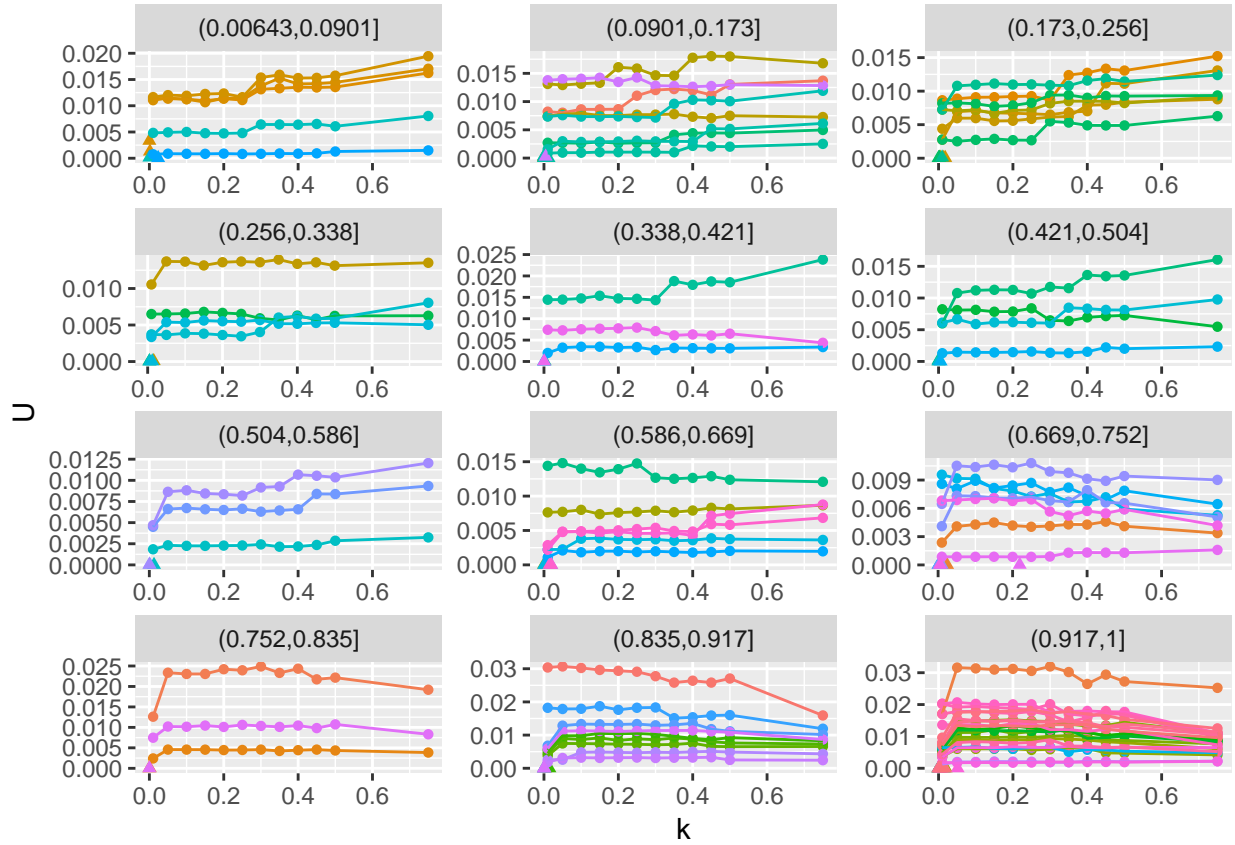


figura 13. $U \sim \%k$ ($\sim p_class$)

m

```
df_resultados %>% ggplot(aes(x = k, y = log(m), colour = SiteCode, shape = modelo)) +
  geom_point(aes(shape = modelo)) + geom_line() + facet_wrap(~p_class, ncol = 3,
    scales = "free") + theme(legend.position = "none")
```

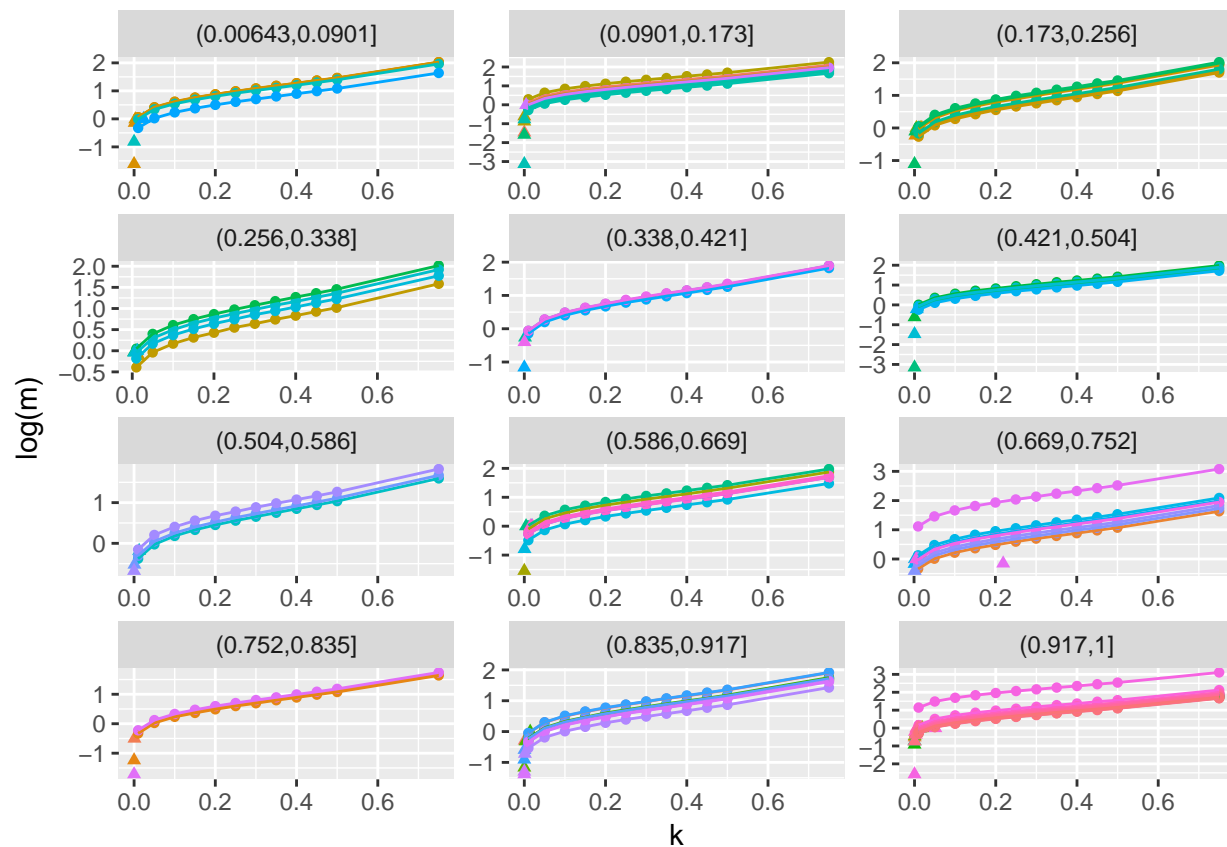


figura 14. $\log(m) \sim \%k$ ($\sim p_class$)

```
df_resultados %>% ggplot(aes(x = k, y = m, colour = SiteCode, shape = modelo)) +
  geom_point(aes(shape = modelo)) + geom_line() + facet_wrap(~p_class, ncol = 3,
    scales = "free") + theme(legend.position = "none")
```

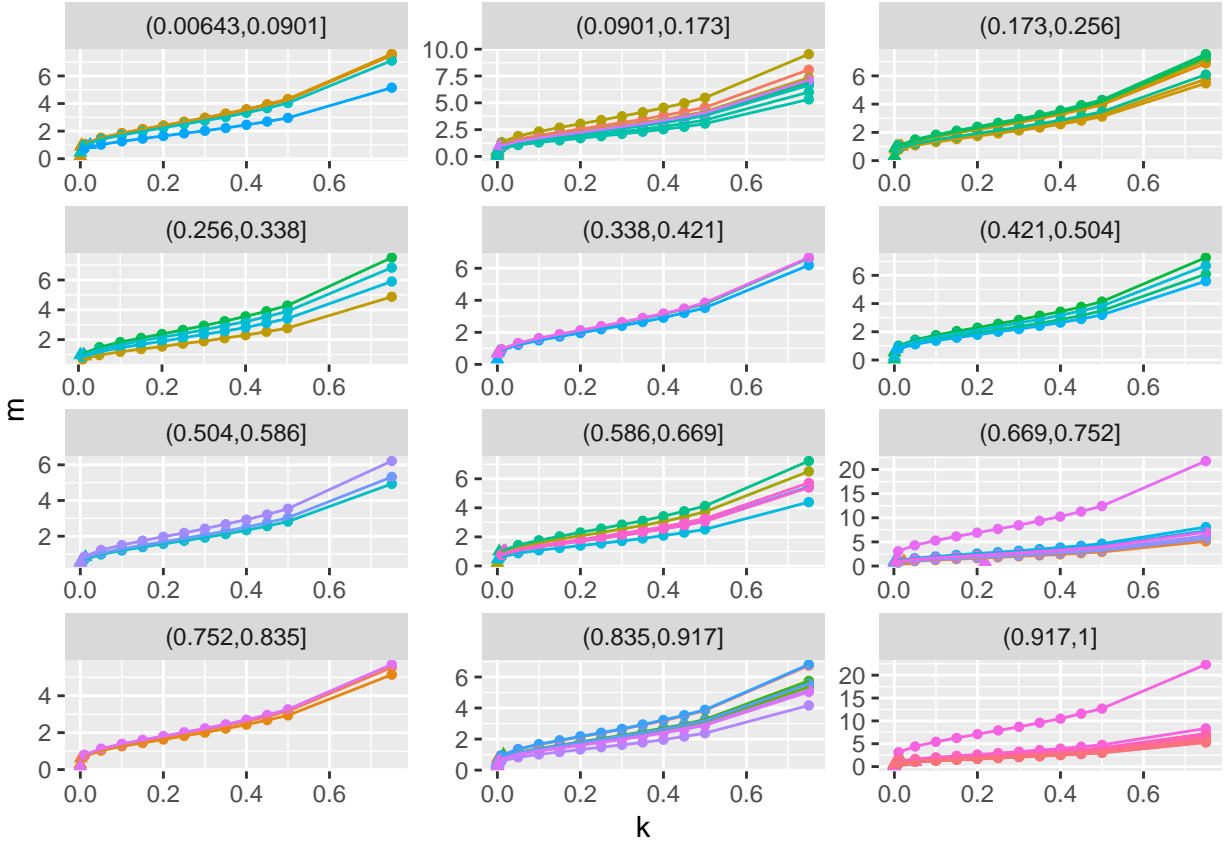


figura 15. $m \sim \%k$ ($\sim p_{\text{class}}$)