

MAP2212 - Laboratório de Computação e Simulação

Relatório - EP05

Ayrton Amaral Alves Vítor - 11288131
Vítor Garcia Comissoli - 11810411

1 Introdução

O objetivo desse EP é, assim como no EP04, a estimação da área ($u(\mathbf{v})$) de uma função dirichlet dado um certo limite superior \mathbf{V} . Para estimar tal área deve-se utilizar uma aproximação por "histograma", ou seja, a área deve ser dividida em um número k de retângulos (tomemos a proporção de $k = \frac{n}{2}$ obtida no EP anterior) no eixo y (retângulos empilhados uns acima dos outros na horizontal ao invés de lado a lado na vertical), denominados bandas (ou bins). Para isso deve-se somar as áreas de todas as k bandas (bins) geradas.

Entretanto, diferentemente do EP anterior, deve-se obter a função geradora aleatória da distribuição dirichlet por meio de um algoritmo de MCMC (Monte Carlo Markovian Chains), mais especificamente por Metropolis - Hastings.

Além disso, deve-se obter um erro de, no máximo, 0.05%, com 95% de confiança, ou seja, que fique dentro do Intervalo de Confiança $[u(v) - 0.0005, u(v) + 0.0005]$ com $\alpha = 0.95$.

2 Discussão Teórica

Como esse EP é um complemento do EP04, a primeira parte da discussão teórica seria a mesma, por isso optamos por não nos repetir. Com isso, na parte teórica será discutido o método utilizado para a função geradora, conhecido como *Monte Carlo Markov Chain method* (**MCMC method**) e o valor de n necessário para atingir a precisão solicitada.

2.1 Método MCMC

O método MCMC consiste, basicamente, na aproximação de uma distribuição qualquer por meio de uma distribuição estacionária (Π) de uma cadeia de markov criada especificamente com esse intuito. Nesse EP, a distribuição que queremos é uma dirichlet, e foram-se utilizadas amostras obtidas a partir de uma distribuição Normal bivariada para criar e aperfeiçoar tal cadeia.

O aperfeiçoamento se dá pelo número de iterações que a cadeia de Markov passa até atingir a distribuição estacionária, já que, utilizar uma cadeia com um número de iterações menor que o número necessário para que se atinja a estacionária não dará uma estimativa tão precisa quanto uma cadeia que já atingiu tal distribuição.

Diferente do método de Aceitação e Rejeição, em MCMC, os valores tendem a ficar mais próximos da região de maior probabilidade da distribuição desejada, já que o n -ésimo valor obtido pode, ou permanecer com a mesma amostra sendo utilizada como média para a distribuição à priori (nesse caso uma Normal bivariada) caso a nova amostra seja menos provável que a anterior (com probabilidade de permanência inversamente proporcional ao quão provável é aquela nova amostra obtida), ou mudar (com probabilidade 1) para uma nova amostra caso ela tenha um probabilidade maior que a anterior.

Vale ressaltar que esse método demora algumas iterações para se aproximar da distribuição desejada, então deve-se descartar um certo número de amostras iniciais que, muito provavelmente, não correspondem

à distribuição desejada (também conhecido como burn-in). Para este fim, descartamos as primeiras 1000 iterações.

2.2 Obtenção Amostral de \mathbf{n}

Fixando os valores do input (\mathbf{x} , \mathbf{y} e \mathbf{v}) e tomando $k = \frac{n}{2}$, criamos uma função que utilizou-se do desvio padrão de vários valores de \mathbf{n} , até que fosse encontrado um valor que garantisse um erro (desvio padrão) ≤ 0.0005 .

Realizou-se então a média do teste acima para diferentes valores de x , y e z fixados, onde x , y e $z = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

Disso, obteve-se que $n = 280000$ é o menor valor para \mathbf{n} testado que garante o erro esperado. Assim, tomaremos \mathbf{n} como 280000.

Como para $n = 28000$, $k = \frac{n}{2}$ se mostrou coerente com o erro de 0.05%, temos que $k = 140000$ é um valor suficientemente preciso para \mathbf{k} . Tomaremos assim \mathbf{k} como 14000.

Assim, encontramos um valor para \mathbf{n} (utilizando a mesma proporção para \mathbf{k} obtida no EP04) que satisfaz o intervalo de confiança do erro estipulado pelo exercício.

3 Discussão do Código

O código da classe Estimador consiste de duas funções de classe (`__init__` e `U`) e 3 funções adicionais (`amostra_MCMC`, `potencial_dirichlet` e `gera_v`).

A função `amostra_MCMC` recebe as informações apriori dos parâmetros da Dirichlet e devolve um array com n potenciais de uma dirichlet.

A função `potencial_dirichlet` simplesmente computa o valor do potencial para uma Dirichlet num ponto em \mathbb{R}^3 com os parâmetros dados.

A função `gera_v` recebe \mathbf{k} e os valores simulados da Dirichlet, os ordena e divide em k bins, então retorna o valor máximo de cada bin.

Ao instanciar um objeto Estimador, nos utilizamos das funções anteriores para obter os valores da `amostra_MCMC`. Então estimamos o integral pela função `U`, que computa a razão entre os valores que são menores que o valor \mathbf{v} dado e o número total de valores gerados.

4 Resultado

Mesmo com o cálculo do n demonstrado acima, por algum motivo que não conseguimos identificar, o código aparenta não condizer com o resultado que lhe era esperado, assim não atingindo a precisão estipulada pelo enunciado do EP.

Como não encontramos o erro no código do EP, foi decidido enviar aquilo que foi feito, visto que mostra a lógica utilizada na realização do mesmo.

Já quanto a estimativa $u(\hat{v})$, pode-se dizer que ela, idealmente, se aproximaria ao valor de real de $u(v)$, e que, com um valor de \mathbf{n} suficientemente grande, o erro da estimativa tenderia a 0.

5 Conclusão

Pode-se concluir então que o programa infelizmente não atingiu os objetivos que lhe foram propostos, já que os valores obtidos pelos testes acima não se mostraram coerentes com aquilo que era esperado pela parte teórica (pela obtenção de um valor para \mathbf{n} que assegura o intervalo de confiança de erro proposto para $\mathbf{u}(\mathbf{v})$), e com aquilo que foi solicitado no enunciado do EP (obtenção de $u(\hat{v})$).

Um elemento notável porém é o fato da geração de números pela Dirichlet é muito mais eficiente que pelo método de aceitação rejeição, pois a densidade da função auxiliar muda a cada nova estimativa.