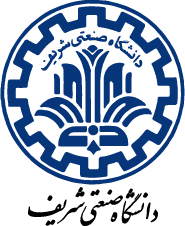
**به نام خدا**



**تمرین takehome**

**برنامه­ریزی تصادفی**

**احمد امامی**

**99207521**

فهرست مطالب

[شرح مسئله 3](#_Toc94298051)

[پارامترهای مدل 3](#_Toc94298052)

[متغیرهای تصمیم مدل 4](#_Toc94298053)

[تابع هدف 4](#_Toc94298054)

[محدودیتها 4](#_Toc94298055)

[حل به کمک الگوریتم کاهش سناریو (پسرو) 4](#_Toc94298056)

[پیادهسازی الگوریتم کاهش سناریو در پایتون 5](#_Toc94298057)

[توضیحات کد الگوریتم کاهش سناریو 6](#_Toc94298058)

\*\* تمام کد­ها و خروجی­های مرتبط با هرکدام در فایل نوت­بوک پایتون، بدون اینکه نیاز به ران کردن مدل داشته باشید قابل مشاهده است.

**مسئله­ی تصمیم­گیری کشاورز را در حالت چند مرحله­ای در نظر بگیرید. برای حالت­های زیر مسئله را نوشته و جواب بهینه را به دست بیاورید:**

**1. تابع توزیع پیوسته برای متغیر­های تصادفی و استفاده از روش حل scenario reduction**

**2. تابع توزیع گسسته برای متغیر­های تصادفی و استفاده از روش حل تجزیه­ی تو در تو**

**جواب­های بدست آمده از دو روش را با یکدیگر مقایسه کنید. برای این منظور نیاز است تعداد مراحل در هر دو حالت مشابه و ارتباط معناداری بین توابع توزیع و متغیر­های تصادفی وجود داشته باشد.**

# شرح مسئله

همانطور که در مسئله­ی کشاورز صفحه­ی 4 کتاب توضیح داده شده است، کشاورزی میخواهد سه محصول گندم، ذرت و چغندرقند را در زمینی به مساحت 500 هکتار و در طول دو سال متمادی کاشت و برداشت نماید. هدف این کشاورز اینست که ضمن در نظر گرفتن محدودیتهای دنیای واقعی طوری زمین زیر کشت خود را به کاشت این سه محصول تخصیص دهد که هزینه­هایش کمینه شود.

برای حالت سه مرحله­ای که در صفحه­ی 18 کتاب درسی بیان شده است، ابتدا مدل مربوطه را می­نویسیم:

همان طور که در صورت سوال نیز مطرح شده است، مرحله­ی اول در این مدل به کشت محصول در سال اول اختصاص دارد. در مرحله­ی دوم تصمیم گرفته می­شود که با مقادیر برداشت شده از کشت اول چه باید کرد و چه مقدار باید فروخته و چه مقدار باید خریداری شود. در این مرحله هم­چنین مقدار کشت هر محصول برای برداشت در سال دوم نیز تعیین می­شود. در مرحله­ی سوم و پایانی نیز در ارتباط با محصولات کشت دوم تصمیم­گیری می­شود.

|  |  |  |
| --- | --- | --- |
| T=3  مرحله­ی سوم | T=2  مرحله­ی دوم | T=1  مرحله­ی اول |
| تصمیم در ارتباط با برداشت سال دوم | تصمیم در ارتباط با برداشت سال اول و کاشت محصول برای سال دوم | کاشت محصول (سال اول) |

با توجه به توضیحات مطرح شده مدل مسئله­ را می­نویسیم:

## پارامتر­های مدل

|  |  |
| --- | --- |
| میزان برداشت محصول i در سناریوی s |  |
| احتمال رخداد سناریوی s |  |
| هزینه­ی کاشت هر هکتار از محصول i |  |
| قیمت فروش محصول i (برای چغندر قند قیمت بالاتر را و قیمت پایین­تر را با نمایش می­دهیم |  |
| قیمت خرید هر تن محصول i |  |
| حداقل کشت مورد نیاز از محصول i |  |

## متغیر­های تصمیم مدل

|  |  |  |
| --- | --- | --- |
| میزان زمین اختصاص داده شده به کشت برای محصول i |  |  |
| میزان خرید گندم مرحله­ی t و تحت سناریوی s |  |  |
| میزان خرید ذرت مرحله­ی t و تحت سناریوی s |  |
| میزان فروش گندم در مرحله­ی t و تحت سناریوی s |  |  |
| میزان فروش ذرت در مرحله­ی t و تحت سناریوی s |  |
| میزان فروش چغندرقند با قیمت بالا در مرحله­ی t و تحت سناریوی s |  |
| میزان فروش چغندرقند با قیمت پایین در مرحله­ی t و تحت سناریوی s |  |

## تابع هدف

## محدودیت­ها

# حل به کمک الگوریتم کاهش سناریو (پسرو) و تولید سناریوفن

همان­طور که می­دانیم پارامتر غیرقطعی که سناریو­ها را به کمک آن تعریف می­کنیم می­باشد. ابتدا باید توزیع پیوسته­ای را برای این پارامتر در نظر بگیریم. در این مسئله فرض کردیم که این مقادیر از توزیع نرمال پیروی می­کنند. در حالت گسسته مقادیر برداشت عادی برای این سه محصول به ترتیب برابر با 2.5 ، 3 و 20 می­باشد. اگر این مقادیر را میانگین توزیع در نظر بگیریم میتوانیم توزیع نرمالی را برای هر کدام در نظر بگیریم. فرض می­کنیم که هر یک توزیعی به شکل زیر داشته باشند:

سناریو­های تولید شده توسط توابع پیوسته­ی بالا بسیار زیاد هستند در نتیجه نیاز داریم تا آن­ها را به صورت گسسته تقریب بزنیم و به کمک کاهش سناریو، سناریو­های محدودی را به منظور حل نهایی بکار ببریم. برای تشکیل سناریوفن بازه­ی را به 100 قسمت مساوی تقسیم می­کنیم. پس از تقسیم، به منظور پیش­بردن محاسبات نیاز است که احتمال هر بخش را حساب کنیم. وسط هر یک از این 100 بازه­ را به عنوان یک سناریو در نظر می­گیریم. نتایج حاصله را در یک فایل اکسل قرار داده­ایم و سپس به کمک آن الگوریتم کاهش سناریو را می­نویسیم.

# پیاده­سازی الگوریتم کاهش سناریو در پایتون

برای پیاده سازی الگوریتم از پایتون و کتابخانه­ی docplex استفاده نموده­ایم. این کتابخانه امکانات متعددی را در اختیار دارد که به راحتی می­توان بدون نیاز به نرم افزار cplex، مسائل برنامه­ریزی را در محیط پایتون کد کرده و حل نمود. کد الگوریتم را در شکل زیر مشاهده می­کنید. لازم به ذکر است که الگوریتم ذیل برای کاهش سناریو­ها به 10 عدد نوشته شده است.

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import scipy.stats as scs

from math import  inf

scenarios=pd.read\_excel('scenarios.xlsx',header=None)

#making a loop to calculate the distance between each scenario

#and add it to the distance matrix defined in the previous cell

#create a dataset of our scenarios

main\_scenarios=scenarios.iloc[:,:3]

#create a dataset of probabilities

probabilities=scenarios.iloc[:,5]

#define a matrix which will contain the distances of scenarios from each other

distance\_matrix=np.zeros([100,100])

probabilities=np.array(probabilities)

minimum=0

epsilon=0.065

while minimum<epsilon:

    for i in range(distance\_matrix.shape[0]):

        for j in range(distance\_matrix.shape[1]):

            if i==j:

                distance\_matrix[i][j]=inf

            else:

                a=np.array(main\_scenarios.iloc[i])

                b=np.array(main\_scenarios.iloc[j])

                dist=np.sqrt(np.sum((a-b)\*\*2, axis=0))

                distance\_matrix[i][j]=dist\*probabilities[i]

    minimum=min(distance\_matrix.reshape(distance\_matrix.shape[0]\*\*2,))

    for i in range(distance\_matrix.shape[0]):

        for j in range(distance\_matrix.shape[1]):

            if distance\_matrix[i][j]==minimum:

                row=i

                column=j

                break

    #update scenarios

    minimum=distance\_matrix[row,column]

    probabilities[column]=probabilities[row]+probabilities[column]

    probabilities=np.delete(probabilities,row,axis=0)

    main\_scenarios=main\_scenarios.drop(row).reset\_index().drop(["index"], axis=1)

    distance\_matrix=np.delete(distance\_matrix,row,axis=0)

    distance\_matrix=np.delete(distance\_matrix,column,axis=1)

    if len(main\_scenarios)==10:

        break

#let's see the new scenarios

main\_scenarios

# توضیحات کد الگوریتم کاهش سناریو

ابتدا فایل اکسل سناریو­ها را در پایتو فرا می­خوانیم و آن را در متغیری ذخیره می­کنیم. سپس سه ستون اول دیتاست مربوطه را در متغیری جدا ذخیره می­کنیم. احتمالات این سناریو­ها را نیز در متغیری جداگانه ذخیره می­کنیم. پس از آن یک ماتریس صفر که از 100 سطر و ستون ( به تعداد سناریو­های تولید شده)تشکیل شده را تعریف می­کنیم. این ماتریس، **ماتریس فواصل** ما خواهد بود و مقادیر فاصله­ی هر سناریو از سناریو­های دیگر را در آن قرار خواهیم داد. پس از این قسمت شروع به تعریف الگوریتم می­کنیم.

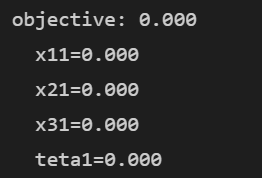
ابتدا مقداری را برای اپسیلون در نظر می­گیریم. این مقدار برای بررسی شرطی است که اجازه ندهیم حداقل فواصل بین سناریو­ها از حد مشخصی بیشتر شود. در صورتی که حداقل فاصله بین سناریوها از این مقدار عبور کند و بیشتر شود دیگر سناریویی را حذف نخواهیم کرد. برای نوشتن الگوریتم نیاز به چند حلقه­ی تو در تو داریم. ابتدا و پس از چک کردن شرط کوچکتر بودن مقدار مینیموم فاصله­ها از اپسیلون تعریف شده، به ازای هر سطر و ستون i و j فاصله­ی سناریوی i و j از هم محاسبه شده و پس از ضرب این فاصله در احتمال سناریوی i، به سطر و ستون متناظر در ماتریس فواصل اختصاص می­یابد. هم­چنین مقادیر روی قطر اصلی را بی­نهایت می­گذاریم زیرا فاصله­ی هر سناریو از خودش همواره صفر است و نمی­خواهیم آن را در بررسی­مان دخیل کنیم. این حلقه به همین ترتیب تکرار می­شود تا تمام عناصر ماتریس فواصل تکمیل گردد. سپس مقدار مینیموم فاصله را حساب کرده و در متغیر minimum قرار می­دهیم. سپس اندیس سطر و ستون این مقدار مینیموم را در ماتریس فواصل پیدا می­کنیم و سناریوی مربوط به اندیس صف را از مجموعه سناریو­های موجود حذف می­کنیم. پس از حذف سناریوی حذف شده، احتمال آن به احتمال سناریوی باقی مانده افزوده می­شود. مجددا این حلقه تکرار شده و این بار ماتریس فواصل با ابعاد 99\*99 را خواهیم داشت و مجددا فواصل را حساب کرده و مینیموم را حذف می­کنیم تا به تعداد سناریوی مدنظرمان برسیم. با گذر مینیموم از مقدار 0.065 الگوریتم متوقف می­شود.

# حل به کمک الگوریتم تجزیه­ی تو در تو

در این بخش از همان سناریو­های مطرح شده در کتاب برای حل مسئله استفاده می­کنیم. در نتیجه سه سناریوی کم­باران، متوسط و پرباران را خواهیم داشت. برای سناریوی کم­باران احتمال 0.3، متوسط 0.4 و پرباران 0.3 را اختصاص می­دهیم.

برای اینکه درک بهتری از الگوریتم کد شده در نرم­افزار بدست بیاوریم مسئله را برای گام اول و جهت Dir می­نویسیم:

پس از کد کردن مسئله­ی جواب بهینه به شکل زیر حاصل می­شود که تمام متغیر­ها برابر 0 هستند.



به کمک یک لوپ هر سه معادله را در پایتون حل می­کنیم و نتایج آن به صورت زیر خواهد بود(کد و خروجی هر دو در فایل نرم­افزار قابل مشاهده است):

solution for: NLDS21 objective: 98000.000 y112 = 200.000 y212 = 240.000

solution for: NLDS22 objective: 98000.000 y122 = 200.000 y222 = 240.000

solution for: NLDS23 objective: 98000.000 y132 = 200.000 y232 = 240.000

برای تا نیز به همین ترتیب است و تنها اندیس s در متغیر­های خرید و فروش تغییر خواهد کرد. مانند قبل به کمک یک لوپ این 9 مدل را حل کرده و جواب بهینه­ی آن­ها را در زیر آورده ایم.

solution for: NLDS31

objective: 98000.000

y113 = 200.000

y213 = 240.000

------------------------

solution for: NLDS32

objective: 98000.000

y123 = 200.000

y223 = 240.000

-------------------------

solution for: NLDS33

objective: 98000.000

y133 = 200.000

y233 = 240.000

...

solution for: NLDS39

objective: 98000.000

y193 = 200.000

y293 = 240.000